Multimed Tools Appl (2018) 77:17333–17373
https://doi.org/10.1007/s11042-017-5308-3

An overview of steganography techniques applied to the protection of biometric data

Mandy Douglas[1] · Karen Bailey[1] · Mark Leeney[1] ·
Kevin Curran[2]

Received: 20 January 2017 / Revised: 31 August 2017 / Accepted: 13 October 2017 /
Published online: 29 November 2017

Abstract Identification of persons by way of biometric features is an emerging phenomenon. Over the years, biometric recognition has received much attention due to its need for security. Amongst the many existing biometrics, fingerprints are considered to be one of the most practical ones. Techniques such as watermarking and steganography have been used in attempt to improve security of biometric data. Watermarking is the process of embedding information into a carrier file for the protection of ownership/copyright of music, video or image files, whilst steganography is the art of hiding information. This paper presents an overview of steganography techniques applied in the protection of biometric data in fingerprints. It is novel in that we also discuss the strengths and weaknesses of targeted and blind steganalysis strategies for breaking steganography techniques.

Keywords Steganograpy · Biometrics · Image analysis · Security

✉ Mandy Douglas
mandy.douglas@lyit.ie

Karen Bailey
karen.bailey@lyit.ie

Mark Leeney
mark.leeney@lyit.ie

Kevin Curran
kj.curran@ulster.ac.uk

[1] Institute of Technology, LetterkennyPort RoadCo. Donegal, Ireland

[2] Faculty of Computing and Engineering, Ulster University, Coleraine, Northern Ireland

1 Introduction

Biometric systems allow for convenient identification to take place based on a person's physical or behavioural characteristics. In comparison with conventional token-based or knowledge based systems, they link identities directly to the owners. Moreover, these identities cannot be given up or lost easily. The uses of biometric procedures have evolved rapidly in the past decade and are used in many different areas, such as banking and government agencies, retail sales, law enforcement, health services, and airport/border controls [3]. In recent years, companies such as Apple and Samsung has integrated biometrics into their latest mobile devices, which can now be unlocked with the owner's fingerprint data [43, 64]. One of the main reasons that these biometric mechanisms are gaining popularity is because of their ability to distinguish between an authorized user and a deceptive one [52. At present, fingerprint biometrics are said to be the most common mechanism, as these are convenient to use, and less expensive to maintain in comparison to other systems. However, as the development of these applications continues to expand, the matter of security and confidentiality cannot be ignored. The security and integrity of biometric data presents a major challenge, as many benefits of biometrics may quite easily become impediment. Thus, from the point of view of promoting the extensive usage of biometric techniques, the necessity of safeguarding biometric data, in particular fingerprint data becomes crucial [37]. For example, fingerprint biometric systems contain sensitive information such as minutia points (explained in the next section) which is used to uniquely identify each fingerprint. The use of latent fingerprints is one way that an unauthorized user can access a system. A latent fingerprint can be easily collected as people leave latent prints when they touch hard surfaces. If an unauthorized user was successful in retrieving a latent print it may enable him/her to gain access to the system hence potentially endanger the privacy of users. Additionally, stolen data may be used for illegal purposes, such as identity theft, forgery or fraud. Therefore, increased security of the data is critical [51].

There are procedures in existence that can help to optimize the security of biometric data, one being, information hiding. Information hiding techniques like watermarking and steganography can add to the security of biometric systems. Watermarking can be explained as a process of embedding information into a carrier file in order to secure copyright, typically ownership [58]. Watermarks can be either visible or nonvisible to the human eye. Steganography is the process of hiding critical data (i.e. identity pin) in a trusted carrier medium (i.e. digital fingerprint image) without third parties sharing any awareness that the information exists. Both methods of information hiding are closely connected [24]. Steganography can be applied using the following two approaches: reversible and irreversible [100]. A reversible data hiding technique, allows for a full recovery of the original carrier file even after extraction of the hidden data. Whereas, an irreversible technique may leave the original carrier file distorted after the hidden data is extracted [88]. Over the past number of years, many image-based steganography methods have been broadly classified depending upon the domain as spatial domain steganography and frequency domain steganography. In Spatial domain steganography, methods such as correlation based techniques, pixel value differencing and LSB substitution, which will be explained later, have been developed and tested. Frequency domain steganography methods consist of many different domains, such as Discrete Cosine Transform (DCT) domain, Discrete Fourier Transform (DFT) domain, Discrete Wavelet Transform (DWT) domain, Singular Value Decomposition (SVD). Frequency domain methods are considered to be more robust than that of spatial domain methods [46, 58, 93, 99].

In recent years, frequency domain methods have been used in combination with other techniques, this approach is known as hybrid steganography. Many of these hybrid techniques make use of a mathematical decomposition called the Singular Value Decomposition. SVD is considered to be one of the most valuable numerical analysis tools available, mainly because singular values obtain inherent algebraic properties and provide stability that permits secret data to be hidden without degrading the perceptual quality of an image [60, 107]. We next look at biometric systems & biometric security.

2 Biometric systems & biometric security

Biometric systems are basically pattern recognition systems that function by obtaining unique personal and biological characteristics from a human being for verification purposes. They use physical qualities such as face recognition, hand geometry, fingerprints, iris sequences, and personal attributes such as voice recognition, keystroke and handwriting patterns. The use of biometric recognition includes various privacy perks. For instance, biometrics can exclude the need to be mindful of numerous passwords and pin numbers hence there is no need to remember them. Biometrics can also be used to restrain unauthorized users from gaining access to mobile devices, computers, government buildings, bank machines, places of work etc. Moreover, the same biometric data can be used consistently, for everything. Biometric data can be divided into two categories: physiological features, which include DNA, face, hand geometry, fingerprints, iris and retina, behavioural features, which include signature, gait and voice. A person's behavioural features may change during the course of their life, for that reason regular sampling is necessary. In comparison, physiological biometric data requires much less sampling [53].

Biometric systems can operate in two modes, identification mode or verification mode. Prior to the system being set up, firstly a database of reference data has to be created. The database is used to store all the biometric templates, this process is known as the enrolment process. [126]. The process of enrolment involves collecting biometric samples from the user, samples are then evaluated, processed and saved as a template on a database for future use [116].

Verification systems attempt to determine "Is this person who they say they are?" In verification, sometimes referred to as authentication, the user presents the system with a biometric trait so they can be identified as a specific person. The system then will analyse the trait provided against data already stored in the database associated to the user in order to find a match. If the data provided has a high degree of similarity to the data stored in the database then the user is accepted by the system as being genuine. Alternatively, the user is treated as a fake and will not gain the requested access to the system. Verification system can be labelled as a one to one (1–1) matching system. In comparison, identification mode is different, as it attempts to identify a person or biometric trait unknown to the system. This type of system attempts to determine who the user is or who presented the biometric. Identification systems compare user input with all enrolled templates already on the system. The system will then output the template that is most similar to the user's input. Providing data similarity is above a certain threshold the user input will be accepted, else the input will be rejected and the user will be refused access. Identification system can be labelled as a one to many (1 – n) matching system [53, 79].

A user can be verified or identified determined on - (1) Something they know: e.g. a pin number, a password etc. (2) something they possess: e.g. a passport/drivers licence, a bank card or a key (3) Something they are (a biometric trait): e.g. a fingerprint, iris, face etc. shown in Table 1.

Using things we know and own are two simple approaches that are widely used for verification and identification purposes. To use something we know just requires us to have a good memory, but quite often, things we know can simply be guessed. Something we have may be snatched and can easily be copied and used at a later date. People's biometric traits are the one thing that does not need to be memorized and because these biometric traits are determined by using body parts they cannot be easily stolen, lost or duplicated [53].

2.1 Biometric techniques

There are various biometric techniques that can be used for verification or identification purposes. These characteristics can be separated into two techniques, physical and behavioural. Physiological biometric traits include face, iris, and fingerprint, hand geometry, retina and palm print. Behavioural techniques include signature, voice, gait and keystroke [54].

2.1.1 Face

The facial recognition process works by analysing various components of a person's face using a digital video camera. It measures the structure of the face including the dimensions between eyes, nose and mouth. Each user's facial measurements are stored in the systems database during enrolment process and are used as a comparison when the user positions themselves in front of the camera. This biometric method is currently used in verification only systems and is known to have a high success rate [123].

2.1.2 Fingerprints

Every person's fingerprints are unique, and will always maintain their uniqueness explaining why they have been used for many years for authentication purposes [11]. Ones fingerprint consists of a pattern of ridges and valleys (located on the top of the fingertip). The top layer of skin on a finger contains the ridges while the lower skin particles contain a pattern of valleys. The distinctive types of disjunctions in ridges (minutiae) hold adequate discriminatory data to distinguish between various fingerprints. Ridge bifurcation (the area where the ridge splits) and ridge ending (the area where the ridge ends) are the most important minutiae points due to their uniqueness in each fingerprint. Biometric fingerprint systems operate by the user placing their finger on a small optical or silicon reader. This reader is connected to a computer which in turn sends the information to a database, the system can then determine fingerprint uniqueness

Table 1 Methods of identification

Techniques	Examples	Issues
Things we know	Pin number – password etc	Can be guessed, be forgotten
Things we possess	Passport, bank card etc.	Can be stolen/lost, be copied
Things we are	Face, iris, fingerprints	Non-repudiable authentication

[76–78]. Due to the availability of person's multiple fingerprints data makes fingerprint recognition suitable for large scale systems, consisting of millions of entities. However, large scale fingerprint systems require a vast amount of computer equipment (hardware and software) particularly if operating in identification mode [34].

2.1.3 Retina

A retinal recognition scan, quite often confused with an iris scanner, is a biometric technique that uses the unique features of an individual's retina to verify them. A retinal biometric system functions by analysing the blood vessel region which is positioned behind the human eye see. Scanning includes the use of a low-intensity light source that determines the patterns of the retina to a high level of accuracy. Unlike an iris scanner, it requires the user to take off their glasses, position their eye near to the device, and fixate on an infrared light inside a tiny opening on the scanner. The device requires the user to focus on the light for the time it takes the system to verify their identity, usually around several seconds. Many users have claimed this method of verification to be uncomfortable, however as there is no accepted way that a retina can be replicated, and a deceased person's retina would decay too fast, retina scanning is deemed to be a very accurate and secure method of verification [53].

2.1.4 Iris

Iris biometrics operates by scanning and then analysing the characteristics that are present in the coloured tissue around the eye pupil. This area contains over two hundred particles, for example, rings, freckles and furrows, all of which can be used for data comparison. Every individual's iris is different, even twins do not possess the same iris patterns. Iris scanners use a typical video camera and can function from a distance unlike a retinal scanner. They can read the iris through glasses and has the capability to generate a precise measurement. This enables iris scanning to be used for identification purposes as well as verification [40].

2.1.5 Voice recognition

A voice recognition system uses the vocal differences and speaking habits of individual's to differentiate between them. It especially pays attention to pitch tone and frequency therefore the system will function more accurately when noise is kept to a minimum [40]. Although, voice biometrics is a convenient and portable method of identification (i.e. it can be used to gain access to mobile devices such as smartphones), it also has its disadvantages. For example, a high quality copied recording of a person's voice may result in an unauthorized user gaining access to a personal device and in turn retrieving personal information which could lead to fraud [113].

2.1.6 Signature recognition

A signature includes text that is repeated quite regularly in nature. For example, signing a child's homework, signing our name on a cheque. During the signature biometric process a user signs their signature on paper (known as static mode recognition) or sometimes on a tablet type device that sits on top of a sensor (known as dynamic mode recognition). If the system is operating in static mode the signature is verified by measuring the shape of the signature. If operating in dynamic mode verification takes place by measuring spatial coordinates (x, y), amount of pressure applied and the inclination of the actual signature. The database then compares the given signature to its database records. If the signature is compatible the user is granted access. This method of verification usually takes around 5 s [53]. Dynamic mode signature recognition are quite difficult to duplicate. Whereas, a static representation of a signature, could be easily duplicated by computer manipulation, photocopying or forgery [79].

2.1.7 Hand geometry

Hand geometry biometric systems work by determining various hand measurements. For example, the hand shape, palm size and the finger dimensions. The user places the palm of their hand on the surface and aligns it using the guidance pegs which illustrate the correct area for fingers. The device then checks the database and verifies the user. The characteristics of an individual's hand is un-distinctive therefore appropriate to use for the identification process (one-to-many). As hand geometry is not sufficiently distinctive to allow one-to-many searches it is usually limited to one-to-one systems used to verify a person rather than identify them from a database [2]. At present, a hand geometry scanner is incapable of distinguishing between a living hand and a dead hand therefore if an imposter places a fake hand on the scanner and applies adequate pressure, they may, deceive the system and gain access [28].

2.2 Fingerprint as a biometric trait

Research carried out has indicated that fingerprints have been used as a method of identification, dating back as far as 6000 BC, by the ancient Assyrians and Chinese [11]. During these times, many clay potters used the pattern of their fingerprint to mark their work. Bricklayers in ancient Jericho also used this method by imprinting their thumbprints on the bricks they used to build houses. Although fingerprint individuality was acknowledged, there is no existing proof to state that this method was used extensively within any of the mentioned societies [82]. During the mid-1800's experimental studies discovered two critical features of fingerprints that are still valid today, (1) no two fingerprints are the same, (2) they will not change through the course of a person's lifetime [11]. Soon after these findings, organizations such as Scotland Yard were using fingerprints for criminal identification purposes. Digitization of fingerprints began in the early 1960's, since then automated fingerprint recognition has been used in widely. The late 1990's has seen the introduction of inexpensive hardware devices (fingerprint capturing devices), and fast and reliable matching algorithms. Among the many biometric techniques discussed above, the fingerprint biometric is one of the most popular ones, due to its high accuracy rate, ease of use and standardization. Furthermore, It is inexpensive, fast and easy to setup. In order for fingerprint scanning to work efficiently it generally requires the comparison of various fingerprint features. These features consist of patterns that are combined unique features of ridges, and minutia points, found within a fingerprint pattern [50].

2.2.1 Fingerprint patterns

A fingerprint consists of three basic patterns of ridges, the arch, loop and whorl. An arch can be explained as the pattern where ridges begin from one side of the finger, ascent in the centre which develops an arc, and then exits the finger from the opposite side (see Fig. 1a). A loop can be explained as the pattern where ridges begin at one side of a finger to create a curve, and are inclined to exit in the same way they entered (same side - see Fig. 1b).

As seen above in Fig. 1c, in the whorl pattern, ridges are structured in a circular position around a central spot on the finger. In general, researchers have discovered that relatives frequently share similar fingerprint patterns, which has led to the concept that fingerprint patterns are genetic [16].

2.2.2 Minutia points

The major minutia points in a fingerprint consist of: ridge ending, bifurcation, and short ridge as shown in Fig. 2.

Figure 2 illustrates the point where the ridge stops, which is called the ridge ending. The point where a single ridge splits in two is known as a bifurcation point. (See Fig. 2b). Short ridges, also referred to as dots are the shorter ridges which are somewhat shorter in length than the typical ridge length (see Fig. 2c). As each fingerprint is different, both minutiae points and patterns are considered a critical aspect in fingerprint biometrics, so the system can analyse data efficiently [50].

2.2.3 Minutiae extraction process

There are two primary procedures used to extract minutia data, binary extraction and direct grayscale extraction. This binary approach has been intensively studied and is also the backbone of many current fingerprint recognition systems and will also be used within this work. Therefore, a binary minutiae extraction method will be discussed in detail. This technique can be broken down into 4 steps, (1) Image enhancement (2) Binarization (3) Thinning and (4) Feature Extraction [15].

2.2.4 Image enhancement

Many fingerprint images are obtained using various types of scanners, for example, optical sensor, capacitive sensor or thermal sensor. Quite often, the image quality can be poor; this can be for numerous reasons. For example, a user can be uncooperative and make it difficult to retrieve a correct sample (law enforcement), or the user may have dry/oily hands [32]. Therefore the purpose of fingerprint enhancement is to process the

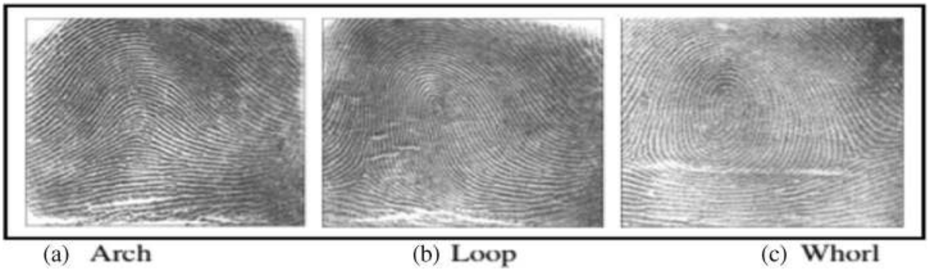

(a) Arch (b) Loop (c) Whorl

Fig. 1 Basic patterns of fingerprint [16]

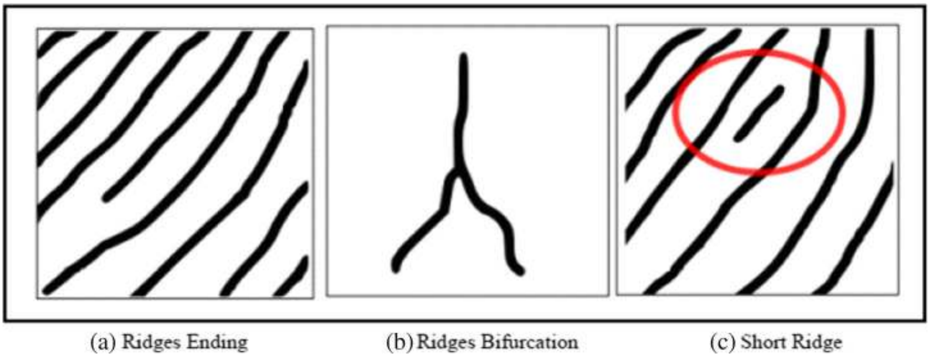


Fig. 2 Minutiae points in fingerprint [16]

obtained fingerprint image in order to upgrade its quality thus make the identification process easier and more accurate [7].

2.2.5 Binarization

During the binarization step the grayscale fingerprint image is converted into a black and white binary image. This procedure is carried out by correlating every pixel value to a threshold value (0.5). If the value of the pixel is lower than the threshold value then the pixel value is assigned black otherwise it is assigned white. The threshold value mentioned here is the default threshold for the MATLAB's 'im2bw' function which will be used for the purpose of binarization in this research. However, it is important to note that other thresholding methods can also be used such as, Otsu's method [110]. After the image is binarized, a process known as thinning is then performed.

2.2.6 Thinning (skeletonization)

Thinning sometimes referred to as skeletonization of the image will reduce the thickness of all ridge lines to one pixel width. It should be noted that this process is quite important as it allows for minutiae to be extracted more efficiently and won't change its location [65]. More on thinning algorithms can be found here ([44, 69]. A sample fingerprint with its corresponding thinned skeleton image is shown in Fig. 3.

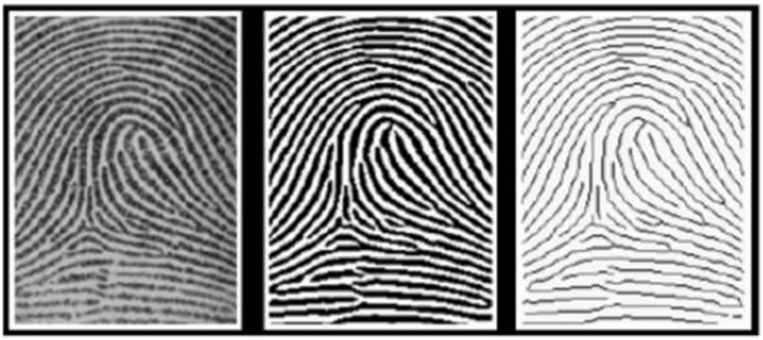


Fig. 3 A fingerprint with its corresponding binary image and ridge skeleton [32].

2.2.7 Minutia extraction

Only a few matching algorithms operate on grayscale fingerprint images directly, therefore an intermediate fingerprint likeness must be derived, this is done during a feature extraction process. A capture device is used to take a distinctive image of the users fingerprint. Distinctive software is then used to examine the fingerprint image and decides if the image truly is a fingerprint, by checking the pattern type (e.g. left loop, right arch, etc.), measuring ridge line qualities, and lastly extracting minutia. Minutiae specify where a significant change has occurred in the fingerprint [10]. These changes are shown in Fig. 4. The dark lines in the image show ridges and the light lines show valleys, Arrow A shows an area where one ridge splits into two (known as a bifurcation) and Arrow B shows where a ridge ends.

When these fingerprint features are located, the extraction software establishes a notable direction of the change (using Arrow B as an example, the notable direction begins at the end of the ridge and progresses in a descending direction). Simply put, the resultant minutia is a group of all reasonable bifurcations and ridge endings, their location, and their specific direction.

2.2.8 Fingerprint matching

Fingerprint matching algorithms work by comparing two given fingerprints and outputs either a percentage of similarity (usually a score between 0 and 1) or a binary decision (match or no match). Only a minority of matching algorithms function directly on grayscale fingerprint images; nearly all of them require that an intermediate fingerprint image be obtained via a feature extraction process [76–78]. A large amount of fingerprint matching techniques can be divided into two families: correlation based and minutiae based. Correlation based matching operates by superimposing two fingerprint images and computes the correlation between corresponding pixels for various alignments (different displacements and rotations). Minutiae-based techniques, which seem to be the most popular approach, extract minutiae

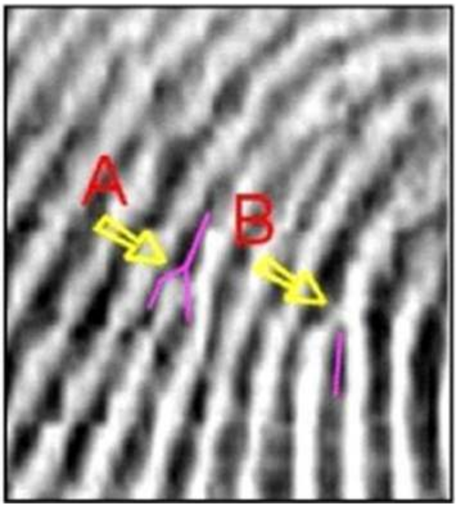


Fig. 4 Fingerprint Changes (fingerprint thesis desktop)

from the two fingerprints and essentially match the alignment between the database template and the minutiae presented by the user shown in Fig. 5.

This approach is deemed an uncomplicated one. However, the binarization and thinning process is believed to be time consuming by some [32]. Therefore many researchers have suggested minutiae extraction techniques that operate precisely on the grayscale images eliminating the need for these procedures [73]. The general concept these authors focused on is tracking the ridge lines within the grayscale image to obtain a polygonal approximation of the ridge line.

2.3 Multibiometric systems

Multibiometric systems identify users by using two or more biometric traits. Research carried out by Patra [84] shows that multibiometric systems are more secure than unimodal biometric systems (biometric systems that rely on only one trait) mainly due to the presence of multiple data. They discuss how a system uses multiple characteristics for authentication purposes and believe that the use of multiple biometrics makes it much more difficult for an intruder to trick the system. Furthermore, a system that uses two or more user traits ensures a live user is present at the time of data acquisition. Multiobiometrics may have improved the security of biometric systems; however security of multi-biometric templates is especially critical as they hold user data regarding multiple traits. If any kind of template data was leaked to an unauthorized person the security and privacy of users may be compromised. [1].

Even though a biometric system can better accommodate users and boost security, they are also vulnerable to numerous types of threats [114]. An imposter may gain entry to the system and browse private data such as medical reports belonging to a genuinely enrolled user. Besides violating user privacy, the intruder can also alter any sensitive information that they have accessed. A genuine user may abuse their authentication rights by entering the system, and maintain that an imposter had done so. For example, a bank employee may alter a customer's bank account details and insist that an imposter could have done this by deceiving the system and stealing the biometric data. An unauthorized user can secretly obtain a user's raw biometric information to gain entry to the system. For example, an intruder may collect an authorized person's latent fingerprint from a specific item, and in time use the fingerprint to create a physical or digital representation of the finger, which in many cases can lead to identity fraud. A biometric user who has access to a wide range of system privileges (i.e. administrator) may intentionally alter system parameters to enable an intruder to attack the

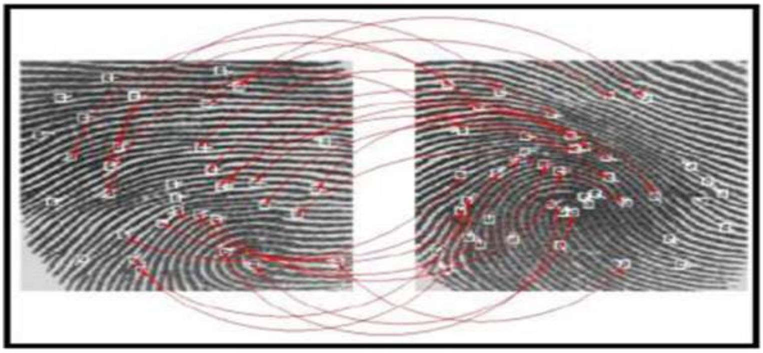


Fig. 5 Matching minutiae points in two fingerprints [16]

system, allowing the intruder to view, change or even steal the biometric data that is stored on the system. An attacker may overload system resources so that genuine users wishing to enter will be denied any service. For instance, a server that deals with access applications can be submerged with an extensive amount of fake requests, thus overloading its data processing resources which would prevent legitimate requests from being processed.

In this section the functionalities of biometric systems were discussed. Various biometric techniques along with their strengths and weaknesses were examined. Fingerprint biometrics was discussed in detail and various feature extraction methods were explored. The weaknesses of biometric systems in regards to security and privacy were also highlighted. Research shows that even though the use of biometrics can boost user accessibility, they are also susceptible to numerous types of attacks. So, in order to enhance the security of these systems, primarily fingerprints, the field of digital steganography will be explored and tested.

3 Steganography

Steganography can be described as the art and science of covert communications which involves the process of hiding information inside other information. Unlike cryptography, steganography messages do not draw attention to themselves, as data is hidden in such a way as to make it undetectable to the human eye. The word steganography is derived from the Greek words “stegos” meaning “cover” and “grafia” meaning “writing”, defining it as “covered writing”. This practice and idea of hiding information can be traced back as far as 440 BC and has been used in many forms over the years [12].

According to Greek historian Herodotus, Histaiacus, a Greek tyrant, used a form of steganography to communicate with his son-in-law Aristagoras. Histaiacus shaved the head of a trusted slave and tattooed a secret message on to his scalp. Once the slave’s hair grew back he was sent to Aristagoras with the hidden message [19]. Another form of steganography occurred in World War 2 when the Germans developed the microdot technique. This method allowed for a lot of information, mostly photographs, to be condensed to the size of a typed period. Information was then hidden in one of the periods on the paper (i.e. a full stop) and distributed over an unprotected channel. The FBI detective, J. Edgar Hoover described the use of microdots as “the enemy’s masterpiece of espionage” [25]. Although steganography has been in existence for many years, its current formation can be explained using the Prisoners’ problem proposed by Simmons [81] where two inmates wish to secretly exchange information to come up with an escape plan. All communication between the two inmates has to pass through a warden. If the warden suspects any type of covert communication has taken place, both inmates will be sent to solitary confinement. All correspondence between the inmates can be checked by the warden, the warden can be either passive or active. If the warden takes a passive approach he\she will attempt to detect if the communication contains any secret information. If covert communication is discovered the warden will make note of it and inform an outside party, information will be allowed to pass through without obstruction. However, if an active warden suspects any hidden information, he/she will attempt to modify the communication by removing or altering the hidden data.

3.1 Steganographic conditions

For a steganographic algorithm to be successful it must adhere to the following requirements: [81]

- *Invisibility:* first and foremost, a steganographic technique needs to be invisible, considering the aim of steganography is to fend off unwanted attention to the transmission of hidden information. If the human eye suspects that information is hidden then this goal is defeated. Moreover, the concealed data may be compromised.
- *Payload capacity* – Dissimilar to the watermarking method of information hiding where only a small amount of copyright data needs to be embedded, steganography aims at covert communication, thus requires adequate embedding space.
- *Robustness against statistical attacks* – Statistical steganalysis is the technique used to discover if hidden information exists. A steganalyst will examine image data by carrying out various statistical tests. Many steganographic algorithms leave a 'signature' when embedding information that can be easily detected through statistical analysis. (Steganalysis will be discussed in more detail in section 5)
- *Robustness against image manipulation* – During the course of the communication process an image can be subjected to changes by an active warden in an effort to expel secret information. Prior to the image reaching its destination it can be manipulated by using techniques such as rotating or cropping. Depending on how the information is embedded, these manipulations may sabotage or ruin any hidden data. A Steganography algorithm is more preferable if it is potent against malicious or unforeseen adjustments to the image.
- *Independent of file format* – As there are an abundance of various image file formats being used on the web, it may attract unwanted suspicion that an individual type of file format is repeatedly communicated amongst two parties. However, if a stenographic algorithm is powerful it should possess the ability to embed data in all types of file formats. This requirement also sorts out the issue of not always being able to acquire a suited image at the correct moment in time, that is, the correct format to use as a cover image.
- *Unsuspicious files* – This requirement contains all features of a stenographic algorithm that may consist of images that are not commonly used and can lead to suspicion. For example, file size that are abnormal may attract suspicion, thus result in further examination of the image by a warden.

An essential condition of a steganographic system is that the image being used (stego-image) for steganography purposes must be as close as possible to the original image, as not to raise suspicion or attract any unwanted attention to the stego image. Image embedding capacity and data invisibility are two primary requirements that have been extensively researched in different steganography techniques over the years [55, 56].

In 1999 [57] presented a thorough survey on 'Information Hiding'. Steganographic methods in use today have progressed a lot since then. In 2006 [8] produced a paper which examined various spatial domain techniques using the least significant bit approach, applied to the GIF image format. Goel and colleagues presented a more recent study on image steganography techniques, published in 2013 [43].

3.2 Digital image steganography

Due to the expansion of the World Wide Web there has been a noticeable increase in the use of digital images. The large quantity of redundant bits that exist within a digital image representation, makes images more preferable for embedding steganographic data. An abundance of diverse image file formats exist within the digital image domain. For each of these different image formats, various steganographic techniques exist [81]. As mentioned earlier, steganography can be implemented using reversible and irreversible techniques. Prior to exploring these techniques, it is necessary to gain an understanding of digital images.

A PC presents images as an assortment of binary digits, comprising distinctive light intensities, in the various image sections [81]. This digit representation constructs a grid. The various locations on the grid are known as pixels. Generally, most digital images on the web are made up of a rectangular graph consisting of images pixels, (bits) where each pixel's colour is contained. These pixels are presented on the grid horizontally, row by row. The bit depth, which also can be explained as the total number of bits in a colour scheme, relate to the total amount of bits used for individual pixels. In Greyscale or Monochrome images, each pixel uses 8 bits and is capable of displaying 256 various colours or shades of grey. Digital images that are coloured normally contain 24-bit files and use the RGB colour model. The bit depth of modern colour schemes is 8; this means that 8 bits are needed to represent the colour of each pixel. All colour variations for pixels of a 24-bit image derive from three colours: red, green and blue, and all colours are represented by 8 bits. Therefore, in one pixel, there can be 256 specific amounts of red, green and blue, producing more than 16-million colours. In addition, the more colours displayed, the larger the image file will be [66].

To transmit an image over the internet successfully it must be an appropriate size. In some cases, (minimum storage, system performance) larger images may not be appropriate, smaller images may be preferred. In certain circumstances, mathematical formulas can be used to decrease the size of the image by condensing the image data, consequently reducing the image size. This technique is known as compression, which can be either lossy or lossless. Both approaches compress the image to save on storage, but are implemented quite differently [13]. The lossy compression technique decreases the file size by eliminating redundant bits of data from the original image. It eliminates areas of the image that are not visible to the human eye; as a result some data may be lost. Although the compressed image bears a close resemblance to the original image, the compressed image is not an exact duplicate, mainly due to data elimination. An example of an image format that uses lossy compression is JPEG (Joint Photographic Experts Group). The JPEG file format will be discussed in detail in the next section [67]. In contrast, lossless compression does not discard any data from the original image. After compression, all original data is restored. This technique would generally be used for spread sheets or text files where loss of data would cause problems. The down-side of this technique is the larger image size. Image formats such as Bitmap, PNG and GIF use lossless file compression [18].

Unlike other information hiding techniques, the main goal of steganography is to ensure that any hidden data is invisible to the human eye. As discussed above, there are many requirements that a steganographic algorithm must satisfy to ensure the secrecy of hidden information. The use of digital images and image compression plays a significant part in choosing which steganographic algorithm to use. For example, lossy compression methods (relating to JPEG images) provide smaller image file sizes, but it intensifies the probability of the hidden information being altered or lost based on the fact that some redundant data is always eliminated. Lossless compression (relating to GIF, PNG images) allows for an image to be compressed without any loss of data, allowing the original image to be maintained. As a result of the lossless approach the image will be larger in size. Lossless image formats may not be suitable for hiding biometric data, as biometric systems also require a fast response time as well as strong security measures [103]. The reversible data hiding technique is another approach to lossless data hiding [100]. Reversible algorithms are often implemented to preserve sensitive images such as medical, government or military imagery. For images like these, even the slightest distortion caused by data embedding is unacceptable [108]. For example, a medical related image such as an x-ray, even a minor change to the image may lead to misinterpretation by a medical practitioner. Hence, why a reversible hiding technique would be a more appropriate approach in this case. Many steganographic algorithms have been developed for both of the above compression techniques and will be explained in detail in the next section.

4 Data hiding in digital images

Two of the most popular digital image formats relating to internet usage are Joint Photographic Experts Group (JPEG) and Portable Network Graphics (PNG). Other image formats are also used, such as Graphics Interchange Format (GIF), but to a lesser degree. Most of the steganographic techniques created were constructed to manipulate the design of the image formats mentioned [21].

Embedding information using steganography can be carried out by inserting the following line of code into a Microsoft command window:

```
C : \ > Copy Cover.jpg/b + Message.txt/b Stego.jpg
```

The above code appends the hidden information found in the text file 'Message.txt' inside the JPEG image file 'Cover.jpg' and constructs the stego-image 'Stego.jpg'. The concept behind this is to exploit the recognition of EOF (End of file), that is, the information is loaded and added after the EOF tag. When observation of the Stego.jpg occurs using any image editing tool, the latter simply exhibits the image disregarding anything that follows the EOF tag. However, if opened in Notepad, the hidden data will be unveiled. The embedded data does not decrease the quality of the image. Image histograms or visual perception will identify any disparity between the two images as the secret data is hidden after the EOF tag. Although this technique is easy to implement, many steganography programs distributed on the internet make use of it (Camouflage, JpegX). Unfortunately, this simple procedure would not withstand any type of altering to the Stego-image nor would it endure steganalysis attacks [90]. Another straightforward method is to affix secret data to the Extended File Information of the image, this is a common approach taken by the manufacturers of digital cameras to store metadata info in the image header file, and i.e. the cameras make and model. However, this technique is just as unreliable as the preceding approach as it is very simple to overwrite such information [20]. In recent years, data hiding, using the LSB embedding method within the spatial domain (pixel level) of images was a very popular technique. This was mainly due to its potentially sizable capacity and its simplicity. More recent studies investigated the frequency domain [12, 46, 104].

Steganography methods can generally be restricted to Spatial Domain, Frequency Domain and Hybrid Techniques.

4.1 Spatial domain techniques

Least significant bit (LSB) replacement is a typical, straightforward procedure for inserting information into a cover image [42]. During this process, the LSB within the cover medium can be overwritten with the binary representation of the secret data. In the case of using a 24-bit colour image individual components are capable of storing 1 bit of d'ata in its LSB. For an example, take the 3 neighbouring pixels (9 bytes) below:

(00101101 00011100 11011100)
(10100110 11000100 00001100)
(11010010 10101101 01100011)

First off, the binary representation 11,001,000 (200), is inserted into the least significant bits of this section of the image; the resulting grid is then as follows:

(0010110**1** 0001110 $\underline{\mathbf{1}}$ 1101110**0**)
(1010011**0** 1100010 $\underline{\mathbf{1}}$ 0000110**0**)
(1101001**0** 1010110 $\underline{\mathbf{0}}$ 01100011)

The binary number was embedded into the first 8 bytes of the grid. However, only 3 existing bits had to be modified (bits are denoted with underline) for the required data to be embedded. Considering there are potentially 256 intensities of each primary colour, modifying the LSB of a pixel results in tiny changes in the intensity of the colours. These changes cannot be recognised by the human eye thus, data hiding the data is accomplished [85]. However, this procedure is especially easy to identify. For example, an attacker looking for uncommon patterns or using various attack techniques (discussed in the next chapter), can quite easily detect any occurrence of hidden information [47]. Additionally, LSB makes use of BMP images, as they use lossless compression. To hide concealed information inside a BMP file would require the cover image to be extremely large. Moreover, BMP images are not often used on the internet and may attract suspicion. For this reason, LSB steganography has also been developed for use with other image file formats [81].

Palette based images, for example Portable Network Graphics (PNG) or Graphics Interchange Format (GIF) images are another common image file format used on the Internet. In recent years, the PNG, format has replaced the older GIF format [127]. Palette based images consist of an index and a palette. The index contains information indicating where each colour is positioned in the palette. It also contains all the colours used in the image and each colour in the palette corresponds to various colour components [81]. Palette based images may also be used for LSB steganography. According to (Johnson) extra care should be taken if making use of this type of format. One issue with the palette approach used with GIF images is that if the least significant bit of a pixel is changed, it may result in creation of, or pointing to an entirely different colour as the index to the colour palette is changed If neighbouring palette entries are alike, there will be no distinct change, but if the neighbouring palette entries are different, the change would be obvious to the human eye [55, 56]. A solution to this problem is to sort the palette so that the colour differences between consecutive colours are reduced [17]. Another solution to this problem would be to use greyscale images for embedding data. An 8-bit greyscale image contains 256 variants of grey thus any changes to the palette may be less noticeable therefore secret data may be harder to detect [55, 56].

Gupta et al. [47] proposed a technique using LSB method by embedding encrypted information into the image in place of plain textual data. The overall process is more complex and time consuming. However, the security of hidden data did improve. [61] also proposed an algorithm to enhance the security of LSB embedding. This embedding procedure also involves an encryption phase. The process involves embedding the secret data into the image using "Least Significant Bit algorithm" by which the least significant bits of the secret document are organized with the bits of a carrier file (digital image). The idea is to merge the message bits with the bits of carrier file. Results show that the proposed approach does improve security and protect secret data from attacks, as data is encrypted and only an authorized person that is aware of the encryption can access the secret information. Tests carried out showed little change to the image resolution and after data was embedded only slight changes occurred in the stego image.

4.2 Pixel value differencing

Another well-known technique used for data hiding in the spatial domain is Pixel Value Differencing [71, 117, 124]. PVD works by calculating the difference between two neighbouring pixels. If the difference output between the two pixels is a large value, then this indicates that the two successive pixels are edge pixels. If the difference value is small, then successive pixels belong to a smooth area of the image. The PVD approach was first introduced by [124] and results showed less distortion to the image when compared with other LSB algorithms. However, this technique does not come without limitation. For example, if the calculated original difference is not equal to the secret data then adjustments need to be made to the two successive pixels, which in turn, may cause much distortion to the stego-image. Wang et al. [117] introduced a method to minimise image distortion by using the modulus function. The modulus function was used to alter the carry-over of the difference between the two successive pixels instead of making an adjudgment to the difference value. In terms of image quality, [117] method showed much better results than the preceding approach implemented by [124].

In a more recent study, [59] proposed a new reversible data hiding technique based on the sorting and prediction of digital images. This proposed technique embeds two bits in a 3 × 1 sub-block at maximum by division of two groups, min and max groups. This method works by firstly predicting the pixel pairs of both the min and max groups. The secret data is then hidden within these predicted pixels. Jung states that reversibility is guaranteed as the order of pixel pairs of the sub-blocks are not changed after embedding secret bits into the to groups. Results showed that the proposed method provides a higher embedding capacity than earlier techniques.

4.3 Transform domain techniques

The following methods attempt to conceal information in the transform domain coefficients of an image. Data embedding in the transform domain is a popular procedure used for robust data hiding. Methods can also realize large-capacity embedding for steganography [46]. According to Goel [42] embedding in the transform domain allows the hidden data to reside in more robust locations, scattered over the entire image. Furthermore, the above techniques also provide greater protection against many types of image processing and steganalysis attacks [86]. To gain an understanding of the above transform domain methods one must firstly describe the sort of file format associated with this domain (JPEG file format). The JPEG file format is the most favoured file format used for data transmission, mainly because of its condensed image size [27].

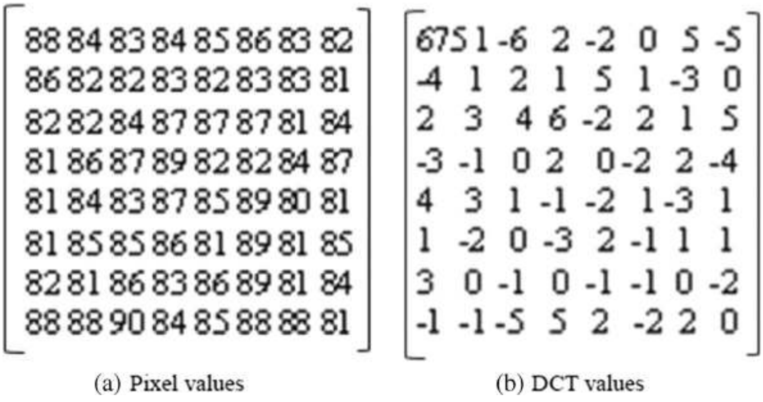
For an image to be compressed into JPEG format, first the RGB colour model must be transformed to a YUV representation. A description of the YUV is as follows: (Y) conforms to the luminance (brightness) of the image, both (U) and (V) conforms to the chrominance (colour). Based on research, the human eye is more delicate to adjustments in the luminance of a pixel than to adjustments to any chrominance. The JPEG compression manipulates this fact by downsizing the colour statistics to decrease the capacity of the file. The colour elements (U) and (V) are split in two in horizontal and vertical ways, hence reducing the size of the file by a component of 2 [26]. The next step is the transformation of the image using the Discrete Cosine Transform.

4.3.1 Discrete cosine transform

When the DCT is applied, the image is divided into parts of differing priorities. It transforms the image from the spatial domain to the frequency domain [43]. This is achieved by organizing image pixels into 8 × 8 blocks and converting the blocks into 64 DCT coefficients. Any adjustment made to a single DCT will alter all 64 pixels within that block [20].

Figure 6 illustrates an example of the application of the DCT to an image and the effects it has on the given image. The left side of the above figure is an 8 × 8 block of image data. Which can be either luminance or chrominance data. The image on the right is the result after the DCT is applied to this block of the image. Notice how the bigger value is positioned in the top-left corner of the block, this is the lowest frequency. The reason this value is very high is because it has been encoded by DCT and the highest priority contains all image energy. Note how all values nearer to the bottom right hand corner are closer to zero, this is because these values contain less energy. These values are classed as the high frequencies; it is these frequencies that will be discarded during the next process [13].

When the image has been transformed quantization is the next stage of the process. During this stage the human eye again is exploited. As discussed earlier the human eye can be



$$\begin{bmatrix} 88 & 84 & 83 & 84 & 85 & 86 & 83 & 82 \\ 86 & 82 & 82 & 83 & 82 & 83 & 83 & 81 \\ 82 & 82 & 84 & 87 & 87 & 87 & 81 & 84 \\ 81 & 86 & 87 & 89 & 82 & 82 & 84 & 87 \\ 81 & 84 & 83 & 87 & 85 & 89 & 80 & 81 \\ 81 & 85 & 85 & 86 & 81 & 89 & 81 & 85 \\ 82 & 81 & 86 & 83 & 86 & 89 & 81 & 84 \\ 88 & 88 & 90 & 84 & 85 & 88 & 88 & 81 \end{bmatrix}$$

(a) Pixel values

$$\begin{bmatrix} 675 & 1 & -6 & 2 & -2 & 0 & 5 & -5 \\ -4 & 1 & 2 & 1 & 5 & 1 & -3 & 0 \\ 2 & 3 & 4 & 6 & -2 & 2 & 1 & 5 \\ -3 & -1 & 0 & 2 & 0 & -2 & 2 & -4 \\ 4 & 3 & 1 & -1 & -2 & 1 & -3 & 1 \\ 1 & -2 & 0 & -3 & 2 & -1 & 1 & 1 \\ 3 & 0 & -1 & 0 & -1 & -1 & 0 & -2 \\ -1 & -1 & -5 & 5 & 2 & -2 & 2 & 0 \end{bmatrix}$$

(b) DCT values

Fig. 6 Pixel Values vs DCT coefficients [13]

sensitive to certain areas of an image. For example, our eyes are relatively good at recognising tiny changes in luminance (brightness) over a relatively large area, however, not so great at recognising various strengths in high frequency brightness. This allows the strength of higher frequencies to be reduced, without modifying the presentation of the image [81] For example, consider an image with a dense collection of trees, in which you have an all-around view. Smaller trees that you don't notice may exist beneath the larger trees in the image. If you cannot see these trees, your view will not be affected if the small trees are there or not. Quantization can be viewed as exactly the same principle. JPEG carries out this process by separating all the values in a block by a quantization coefficient. The outcome is rounded to integer values [13].

The quantised coefficients of the DCT shown above in Fig. 7 are typically normal. There are only a slight amount of individual values where the numbers are larger than zero (most will always be zeros). It is also common practice that all non-zero numbers reside towards the upper left, and zeros to the lower-right corner. Due to the fore mentioned, another process must be applied to group similar frequencies together; this process is called zigzagging. The purpose of this procedure is to group all low frequencies together using a zigzag motion. As stated above, after quantization there will only be a minimal amount of values that hold values (low frequencies) other than zeros (high frequencies), the zig-zag process works by re ordering these values so that related frequencies are brought together. This will allow for high compression to be achieved [13]. See Fig. 8. The final stage uses an algorithm such as Huffman coding to compress the image and Huffman trees are stored in the JPEG header [96].

4.3.2 JPEG steganography

According to [63] it was originally the belief that steganography might not be feasible to use with JPEG images, the reason being, that JPEG's usage of lossy compression. As discussed previously, steganography can make use of redundant bits in an image to embed hidden data, considering redundant bits are omitted in JPEG it was feared that any hidden information would be lost. Moreover, if the hidden information came through unharmed, it may, be equally as challenging to embed information without any adjustments being obvious, due to the severe compression that is used. Nonetheless, attributes of the compression algorithm have been taken advantage of to create a steganographic algorithm for JPEG images labelling the algorithm as being lossy, this attribute too can be used to conceal hidden information [68]. The main advantage DCT has over alternative transforms is its capability to decrease the block-like presentation resulting when the boundaries between the 8 × 8 sub-images become apparent. A disadvantage of DCT being that it only can operate on JPEG files as it presumes a certain numerical

$$\begin{bmatrix} -415 & -33 & -58 & 35 & 58 & -51 & -15 & -12 \\ 5 & -34 & 49 & 18 & 27 & 1 & -5 & 3 \\ -46 & 14 & 80 & -35 & -50 & 19 & 7 & -18 \\ -53 & 21 & 34 & -20 & 2 & 34 & 36 & 12 \\ 9 & -2 & 9 & -5 & -32 & -15 & 45 & 37 \\ -8 & 15 & -16 & 7 & -8 & 11 & 4 & 7 \\ 19 & -28 & -2 & -26 & -2 & 7 & -44 & -21 \\ 18 & 25 & -12 & -44 & 35 & 48 & -37 & -3 \end{bmatrix}$$

(a) DCT coefficients

$$\begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

(b) Quantisation Table

$$\begin{bmatrix} -26 & -3 & -6 & 2 & 2 & -1 & 0 & 0 \\ 0 & -3 & 4 & 1 & 1 & 0 & 0 & 0 \\ -3 & 1 & 5 & -1 & -1 & 0 & 0 & 0 \\ -4 & 1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

(c) Quantised DCT coefficients

Fig. 7 Quantisation procedure [13]

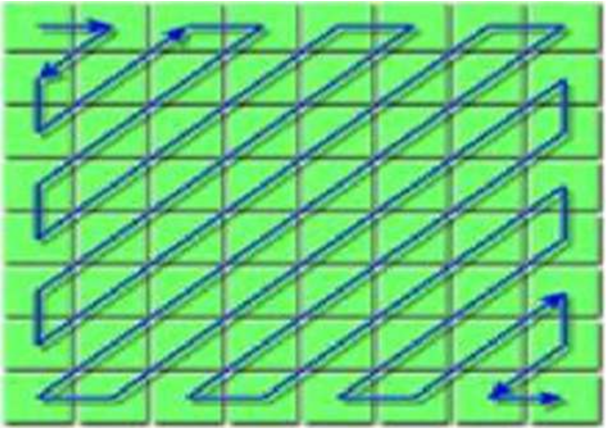


Fig. 8 The Zigzag grouping process [13]

arrangement of the cover data that is generally established in JPEG files. A few common DCT based information hiding techniques are JSteg, F5 and OutGuess [14]. Yet Another Steganographic Scheme (YASS) is an additional method related to JPEG steganography [106].

4.3.3 Discrete wavelet transform

Recently, the Discrete Wavelet Transform (DWT) has proved to be the preferred area of study in the field of information hiding [46, 93, 99]. This is mainly due to its extensive utilization in the new image compression standard, JPEG2000 [41], and its ability to address capacity and robustness [6]. Unlike the DCT procedure, DWT provides frequency, along with spatial description of an image. For example, if the signal is embedded, it will affect the image in a local way. Wavelet transform is believed to be more applicable to data hiding as it divides high-frequency and low-frequency information based on the pixel-by-pixel basis [20]. The DWT divides pixel values into various frequency bands known as sub bands. Each sub band can be described as the following: [12].

- LL – Horizontally and vertically low pass
- LH – Horizontally low pass and vertically high pass
- HL - Horizontally high pass and vertically low pass
- HH - Horizontally and vertically high pass

As mentioned previously the human eyes are much more sensitive to certain areas of an image such as low frequency bands (LL sub- band). This enables information to be hidden in the other three sub bands without any alterations being carried out in the LL sub-band. Each of the other three sub-bands contains irrelevant information as they are high frequency sub-bands. In addition, embedding private information within these sub-bands will not have a big effect on degrading image quality [104]. To gain a better understanding as to how wavelets work the 2-D Haar wavelets will be discussed. A 2-dimensional Haar-DWT consists of two operations, a horizontal and a vertical one. Operation of a 2-D Haar [22] is as follows:

Step 1: First, the pixels are scanned from left to right, horizontally. Next, the addition and subtraction operations are carried out on adjacent pixels. Then, the sum is stored on the left and the difference stored on the right as shown in Fig. 9. The above process is repeated until all the rows are processed. The pixel values sums represent the low frequency element (denoted as symbol L) while the pixel differences represent the high frequency elements of the original image (denoted as symbol H).

Step 2: All pixels are scanned from top to bottom in vertical order. Next, addition and subtraction operations are carried out on adjacent pixels, the sum is then stored on the top and the difference is stored on the bottom as shown in Fig. 10. Again, the above process is repeated until all columns are processed. Lastly, we will be left with 4 sub-bands denoted as LL, HL, LH, and HH. Note, the LL sub-band is the low frequency section therefore looks almost identical to the initial image.

The entire process explained above is called the first-order 2-D Haar-DWT. The effects of applying first-order 2-D Haar-DWT on the image "Lena" is shown in Fig. 11.

In comparison to DCT, recent studies have shown that wavelets are considered as being less resource intensive and cause less distortion to an image hence why the DWT method is becoming a more popular. Moreover, as DWT is broken down into sub-bands, it gives higher flexibility in terms of scalability [31].

4.3.4 Hiding biometric data

Shejul and Kulkarni [104] propose a steganography method based on biometrics. The biometric feature used to implement steganography is the skin tone region of images. The technique suggested involves embedded data in skin region of images. Prior to embedding, the skin tone detection is carried out using HSV (Hue, Saturation and Value) colour space. Additionally, data embedding is implemented using frequency domain approach - DWT (Discrete Wavelet Transform). Secret data is embedded in one of the high frequency sub-bands of DWT by tracing skin pixels in that sub-band. Their analysis shows that by adopting an adaptive technique, in the sense that, skin tone objects are traced in image by cropping various image regions to embed that data, enhanced security is achievable. A skin tone detection steganography algorithm is proposed by [20], which demonstrates robustness to attacks, while keeping the secret data invisible, by embedding in skin regions of an image. This technique is very appropriate for hiding biometric data, especially where templates contain a lot of skin attributes (i.e. facial or fingerprints).

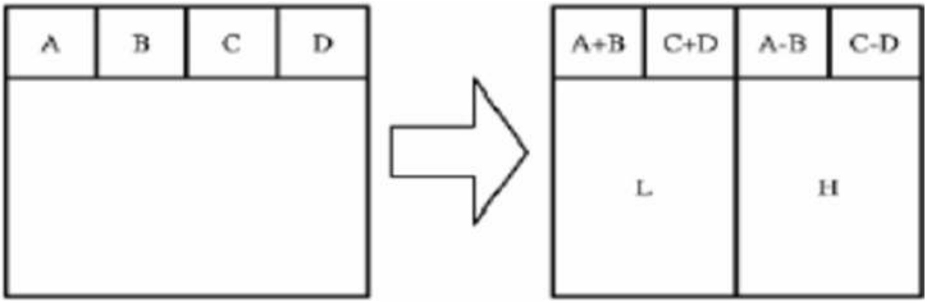


Fig. 9 The horizontal procedure based on the first row [22]

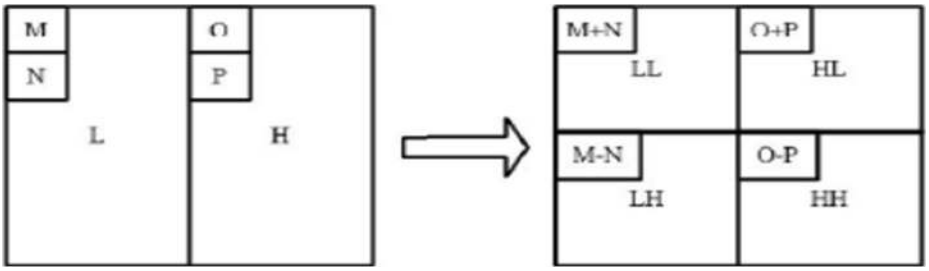


Fig. 10 The vertical procedure [22]

Lavanya et al. [70] introduced a new high capacity Steganography method relating to biometrics. A skin tone detection algorithm is again proposed. Skin tone regions are detected by HSV (Hue, Saturation and Value) colour space and data is embedding in one of the high frequency sub-bands using the DWT transform domain. The embedding process is carried out over a whole block rather than in the image bit planes to provide a secure data embedding location. The authors states that the latter approach ensures that no noisy bit-plane is left unused which will preserve the visual quality of the image.

Amritha and Varkey [4] present a biometric steganographic technique using DWT and encryption. The idea is based on the perception that before secret data is hidden in the cover image it must be encrypted to provide a high degree of security. Again, the skin tone region is the chosen area for data embedding. The proposed application provides invisibility and excellent image quality of the stego image.

Another recent study by [75] examines the security issues of biometric based authentication (fingerprint biometrics). An authentication fingerprint technique is suggested, with steganographic data protection. Malkhasyan puts forward a technique to embed hidden data in the form of a small label into the fingerprint image. The label hidden contains information relating to the fingerprint (i.e. minutia). This can improve the security of the fingerprint by prohibiting unauthorized users, as it will be unknown to everyone that hidden data exists within the actual fingerprint. Although, the author does not believe that this technique will fully secure a fingerprint biometric system, it is speculated that it may be more difficult for an intruder to break the system, due to the embedded label in the fingerprint image.

The aforementioned papers used a variation of watermarking and steganography techniques in the attempt to improve the security of biometric data. All techniques proposed in the papers above showed that biometric data (facial and fingerprints) can be made more secure with the

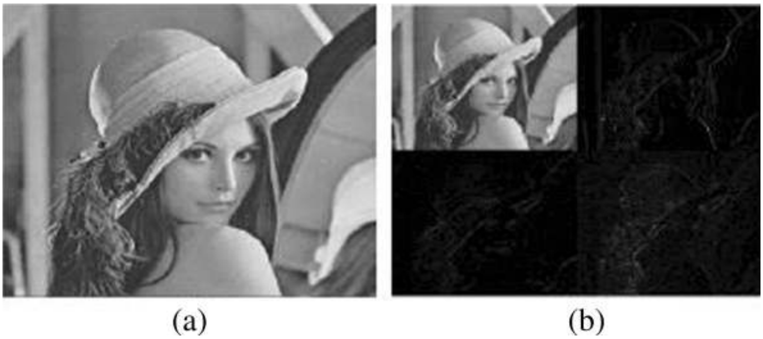


Fig. 11 **a** Original image (**b**) After 2-D Haar DWT is applied [22]

use of various data hiding methods. There are many different ways that data hiding methods can enhance security of biometrics. Consider the following scenario, hiding facial information (e.g., eigen-face coefficients or facial image) within a fingerprint images. Consider that the fingerprint image of a person is stored in a smart card issued to that person (e.g., public service card). When the person uses the card to gain access to secure information or access a secure area, the fingerprint on the person's card will be compared to the fingerprint stored on a database system. The face information secretly hidden within the fingerprint will also be extracted and can be used as a second source of authentication. The facial image can also be verified by using either a biometri c system or by human verification. Hence, by embedding an additional biometric (e.g.,facial image) into another biometric (e.g., fingerprint) using data hiding techniques such as steganography increases the securty of the latter.

In the scenario discussed above, both biometrics were needed to access secure data or system. However, steganography also can be used to secure biometric data in other ways. For example, fingerprint minutiae can be embedded into a carrier image. The carrier image (also known as a host image) sole purpose is to transmit the fingerprint biometrc through a nonsecure communication channel. The carrier is no way connected to the fingerprint data concealed within it. This method of securing biometrics may be useful to government or law enforcment departments to transport data securly from one location to another.

4.4 Hybrid techniques

The aforementioned steganography methods conceal secret data in the spatial or frequency domain. Recent advances in this area show that both security and robustness of a system can be improved by using a combination of two or more of these techniques [115]. This approach is known as a hybrid technique [105]. In recent years, singular value decomposition has been explored and merged with other frequency domain techniques for data hiding in digital images [48, 74, 89]. The literature relating to the above method shows very promising results, especially in regards to image quality and robustness against various attacks such as, compression, noise etc. Therefore, the singular value decomposition will be further investigated.

4.4.1 Singular value decomposition

The Singular Value Decomposition (SVD) is considered to be one of the most valuable tools in linear algebra, with various applications in image compression, data hiding, and many other signal processing areas. If A is an nxn matrix, then SVD of matrix A can be defined as follows: [5]. Note T is used to denote the transpose of the matrix.

$$A = U^{*}S^{*}V^{T}$$

Where U is an mxm orthogonal matrix, V is an nxn orthogonal matrix, and S is an mxn matrix made up of diagonal elements which represents the singular values of the image [98] (Fig. 12).

The columns of the orthogonal matrix U are known as the left singular vectors, and columns of the orthogonal matrix V are known as right singular vectors. The left singular vectors of A are eigenvectors of AA^{T} and the right singular vectors of A are eigenvectors of $A^{T}A$. Each singular value (SV) represents the image luminance, while the corresponding pair of singular vectors represents the image geometry [39]. U and V matrices can be explained

$$\mathbf{SVD(A)} = \begin{bmatrix} | & & | \\ \mathbf{u}_1 & \cdots & \mathbf{u}_\rho \\ | & & | \end{bmatrix} \begin{bmatrix} s_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & s_\rho \end{bmatrix} \begin{bmatrix} - & \mathbf{v}_1 & - \\ & \vdots & \\ - & \mathbf{v}_\rho & - \end{bmatrix}$$

Fig. 12 The SVD operation SVD (A) = U S V_T

further as unitary orthogonal matrices (the sum of squares of each column is unity and all the columns are uncorrelated) where diagonal elements of S satisfy the following properties

$$\sigma_1 \geq \sigma_2 \geq \ldots \sigma_r \geq \sigma_{r+1} \geq \ldots = \sigma_n = 0$$

As an example to clarify SVD transform, consider:

$$\mathrm{A} = \begin{bmatrix} 12 & 23 & 17 \\ 34 & 11 & 25 \\ 18 & 53 & 29 \end{bmatrix}$$

If SVD is applied on the above matrix A, A will be decomposed into the corresponding three matrices as follows:

$$\mathrm{U} = \begin{bmatrix} -0.3970 & 0.0600 & -0.9158 \\ -0.4667 & -0.8724 & 0.1452 \\ -0.7903 & 0.4851 & 0.3744 \end{bmatrix}$$

$$\mathrm{S} = \begin{bmatrix} 77.9523 & 0 & 0 \\ 0 & 27.5619 & 0 \\ 0 & 0 & 1.3349 \end{bmatrix}$$

$$\mathrm{V} = \begin{bmatrix} -0.4472 & -0.7332 & 0.5122 \\ -0.7203 & 0.6347 & 0.2798 \\ -0.5303 & -0.2439 & -0.8120 \end{bmatrix}$$

Here the diagonal components of matrix S are singular values, notice that these values satisfy the non-increasing order: 77.9523 > 27.5619 > 1.3349 [93].

4.5 Properties of SVD

In general, a real matrix (i.e. matrix A above) contains many SV's. Many of these singular values are very small, and the number of SV's that are non-zero equals the rank of matrix A. SVD holds a multitude of good mathematical features therefore; utilization of SVD within the digital image domain has many benefits [98]. For example,

- Large portion of the image signal energy can be represented with very few singular values.
- SV's represent intrinsic algebraic image properties.
- SVD can be applied to square and rectangular images.
- The SV's (singular values) of an image has very good noise immunity, meaning that the image doesn't change significantly after a small perturbation is added.

For example, Fig. 13a, b presents an image and the same image after salt & pepper noise is applied to Lena image. The topmost five singular values of the original image and the salt & pepper image are shown in the Table 2. Notice how the singular values are very similar i.e. the

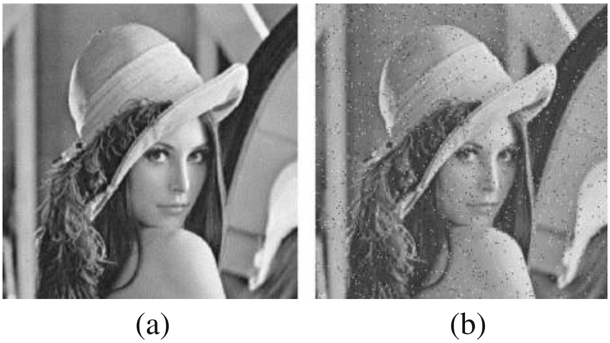


Fig. 13 **a** Original Lena image, (**b**) Salt & Pepper image.

changes in the singular values are minimal hence, providing good stability of the image's singular values, despite manipulation.

4.6 Data hiding schemes based on SVD

Due to the above properties, many data hiding algorithms have been developed and tested based on this method. The main concept of SVD application procedure is to identify the SVD of the cover image and alter its singular values to conceal hidden data. Some SVD techniques are based solely on the SVD domain, i.e. the SVD method is used on its own for the embedding of data; this is known as pure-SVD. However, recent literature has brought to light many hybrid SVD-based techniques which combine various types of transforms domain such as Discrete Wavelet Transform, Discrete Cosine Transform, etc. [38, 48, 107, 112].

A hybrid data hiding technique using DCT and SVD has been presented by [111]. Initially, the DCT is applied on the whole cover image and DCT coefficients are divided into four sections using the zig- zag ordering, then SVD is applied to each section. The four sections mentioned serve as frequency bands from the lowest to the highest. Singular values of the secret image are then used to alter the singular values of each section of the cover image. The approach used in this paper comprises of the cover image being broken down into four parts (blocks) therefore the size of the secret image is equal to quarter size of the cover image. It has been mentioned that concealing information in the lower frequencies bands of the image can aid to robustness against some attacks whereas altering the higher frequencies provide robustness against a different group of attacks, such as noise addition, filtering etc. The authors have carried out tests based on the robustness of this technique against attacks which include JPEG and JPEG 2000 compression, Gaussian noise and blur, histogram equalization, cropping and image rotation etc. Results showed that the algorithm was robust to most attacks. However, the rotation test proved to be unsatisfactory due to loss of embedded data. [38] proposed an SVD-DWT based algorithm, quite akin to the above mentioned technique presented by [111]. They break down the cover image into four sub-bands using DWT and

Table 2 Singular values of two images

Original image	125.5754	20.9756	16.1463	12.8472	11.6251
Salt & pepper image	125.5596	20.9472	16.0555	12.7556	11.5120

SVD is applied to each of the image sub- bands. Then, SVD is applied on the secret image and the singular values of the cover image are altered with the singular values of the secret image. Subsequently, four sets of DWT coefficients are obtained and the DWT inverse is applied which includes the modified coefficients, producing a stego image. The stego image was tested for robustness against various image processing attacks including Gaussian noise, JPEG and JPEG 2000 compression, cropping, histogram equalization, etc. Image quality measure was also tested by comparison of secret data extraction and the original secret data. These test showed no severity to image quality based on the above embedding technique.

A more recent study by Subhedar and Mankar [107] also proposes a technique based on Discrete Wavelet Transform (DWT) and SVD. They embed their secret data using the singular values of the secret image into the cover image based on the modification of the wavelets HH sub-band coefficients. This method also showed very promising results, in relation to many image attacks. Furthermore, after comparison of the stego image against the original cover image, results also look encouraging. The above studies confirm that some SVD hybrid methods have been developed and tested in the area of biometrics, mainly for securing biometric data. However, at the time of this research, very few pieces of literature were found. The present studies in this area seem to focus solely on iris biometric. However, one study proposed a method to improve the authenticity of fingerprint biometrics [9]. [112] proposes an algorithm in order to enhance the security of biometric data (iris template) using DWT-SVD domain. The authors highlight that the integration of the SVD and DWT together produces a more robust and imperceptible strategy for data hiding. They first apply single level DWT to the host image to obtain the set of four sub-band coefficients. This is followed up by application of SVD operation on the high-frequency sub-bands (i.e. HH, HL). Then, a binary representation of the biometric iris template is hidden by modifying the singular values of the high-frequency bands. The inverse of SVD is applied which include the modified SV's. Lastly, the DWT inverse is applied to produce a stego image. The outcome of tests carried out was very encouraging. Image quality tests showed, barely any image distortion after embedding had taken place. Moreover, the method proved to be robust whist analysed against an abundance of popular attacks.

Harmanpreet and Shifali [48] have presented a data hiding technique using a combination of three frequency domains, SVD, DWT and DCT (Discrete Cosine Transform). The projected technique is based on the detection of facial and iris biometric detection, to secure for authenticity and ownership of data. As in prior methods, the wavelet coefficients of the cover image are utilized to embed the secret data; the HH-sub-band is selected for data embedding. Following the DWT decomposition of the cover image, DCT is then applied to the HH band. Subsequently, the SVD application is applied and the singular values of both cover and secret image are retrieved, and added together to produce the modified singular values. Lastly, the inverse DCT transform is applied followed by the inverse DWT. The use of this algorithm for data hiding has proven to be highly imperceptible. Furthermore, it shows robustness against all sorts of attacks, and also possesses very high data hiding capacity. In addition, this technique holds all the requisites required of a model data hiding system such as fidelity, robustness and high capacity. In a study by [9], a robust data hiding algorithm is proposed for the safeguarding of fingerprint images. Again, SVD transform technique is used for embedding secret data. This approach differs from the above techniques as it uses solely the singular value decomposition without any input from DWT, DCT etc. A fingerprint is used as a cover image and a facial image used for embedding purposes. The cover image is divided in to 8 × 8 blocks and the SVD is computed for each block. The diagonal elements of each block, which is the singular values, are then modified with the bit pattern of the secret image content by remainder of the singular values S (1,1) divided by the set value of the image quality 'Q' factor. The inverse of SVD is then applied to produce the new image containing the secret data. The authors mention that various attacks are initiated on the fingerprint images to verify its robustness etc. However, only the outcome of one particular attack (rotation attack) was discussed in the paper. The authors highlight that resistance to rotation is an important factor for fingerprint images yet give no explanation as to why this claim was made. Furthermore, no material was included to verify this statement.

4.7 Conclusion

This section explored current studies in the area of steganography, deployed in spatial domain and transform domains of digital images. In general, a frequency domain approach seems much more attractive than that of a spatial domain, as transform methods (DCT), (DWT) make modifications in the high frequency coefficients rather than directly manipulating the image pixels. Embedding data into the frequency domain causes less distortion to the image, and seems to be a lot more resilient to attacks such as compression, hence why these methods are preferred. In most cases, it is hard to recognise secret data is present, but on the other hand the payload of the hidden information must be small (in comparison to spatial embedding) due to the risk of image distortion, thus a higher possible detection risk. Studies conducted into the field of steganography in biometrics indicate that a frequency domain approach for hiding biometric data is a more preferable approach. The use of low frequency bands often cause the image to become distorted, thus increasing the visibility of hidden data. On the other hand, embedding in the high frequencies also has its downfalls, as attacks such as compression and filtering mainly affect these frequencies. It is likely that embedding data in high frequencies will lead to data disruption, or complete loss of data. A good compromise may be to embed information in mid frequency bands, this may improve, or even solve the above mentioned problems. Even though some negative points, such as small capacity for hiding, have been highlighted in regards to the DWT domain, it still presents a promising outcome and surpasses the DCT domain particularly in surviving compression [94, 118]. In recent years, many hybrid algorithms have been proposed. These techniques are more robust against various image attacks as they utilize the properties of more than one domain. Many of these recent approaches are developed by computing the SVD of a cover image and then modify its singular values to conceal secret data. As the singular values don't much change when small modifications are made, image distortion has been reported as minimal after embedding has taken place, hence less chance of detecting that hidden data is present. Studies show that there are many types of algorithms for data hiding, some of which were discussed above. It is clear that each method has its own advantages and limitations no one method is 100% robust. For example, each technique proved resilient to some type of attacks but showed weakness towards other attack types. It is noticed that more advantages exist in systems using wavelet transforms, such as DWT along with SVD. Many encouraging results have been recorded based on these two domains.

5 Steganalysis

The process of steganalysis can be explained as the art and science of detecting hidden information that occurs through the practice of steganography [49]. Steganalysis is an extremely challenging discipline, as its dependant on vulnerable steganography techniques [83].

According to [36], "the ability to detect secret messages in images is related to the message length". The fore mentioned declaration is established on the sense that if a tiny amount of information is embedded in a sizable carrier file, it will result in a limited percentage of manipulations, thus it will be much more difficult to identify the existence of a concealed communication. There exists two primary classifications of steganography, targeted, and blind [87]. Patil et al. [83] believe that the success of any steganalysis algorithm is dependent on the amount of information the steganalysist has to begin with. Moreover, to successfully attack a steganographic algorithm, a steganalysist must be knowledgeable of the procedures and techniques of many steganography tools [95]. Classification of attacks based on information available to the attacker as discussed by [95] are outlined below:

> *Stego only attack:* In a stego-only attack, only the stego object is available for investigation, the steganalysist does not have any additional information. Realistically, the only way a steganalysist could attack is by trying all common attacks on current steganographic algorithms
>
> *Known cover attack:* In this sequence of events, both the cover object and the stego object are available. As both mediums are available to the steganalyst they can look for variations between the two mediums and therefore can attempt to identify what type of steganographic algorithm was used.
>
> *Known message attack*: In this scenario, the steganalyst is aware of the hidden information, and they can study the stego image for similar future attacks. Sometimes, knowing the message and studying of the stego image help the steganalyst to attack related systems. However, even by knowing the above information, this may still prove to be a difficult task and may even be treated the same as the stego-only attack as the original image is not available for consideration.
>
> *Chosen stego attack:* In this case, both the steganographic algorithm and stego medium (i.e image) are known to the steganalyst. This type of attack may involve the steganalyst attempting to produce stego objects from cover objects in order to pair the seized stego medium. Theoretically, trying to create brand-new stego mediums to pair the seized one seems right, yet in practice it is extremely difficult to achieve, considering both the stego medium and the embedded information is not known

The above classification of steganalytic attacks is rarely used, as the primary objective of steganalysis is to detect the existence, or the absence of concealed information. Most of the current steganalysis attacks were created by the awareness of the algorithm used, just as Kerckhoffs' principle suggests [101, 102], in order to acquire a methodology by constructing stego images with known covers, and thus measure their statistics. As discussed previously, the main goal of steganalysis is to initially detect the existence of hidden information. A more useful list of attacks that are primarily used are the following.

5.1 Targeted attacks

Targeted steganalysis works when a technique planned for detecting a particular steganographic process has been created [83]. For instance, embedding within pixel values leaves behind specific pattern types which can be investigated for with suspicious files. Assuming the steganalyst is confident that secret communications have taken place, and is also aware of an available process as to how the hidden information might me embedded, then it should take only minimum effort to identify whether or not the file consists of this kind of steganography or not. The next few sub sections introduces a few fundamental steganalytical strategies relating to targeted steganalysis, and includes visual, structural, and statistical attacks.

5.1.1 Visual attacks

According to [83] visual attacks are considered as the simplest form of steganalysis. Just as the name implies, a visual attack is generally associated with investigation of the stego object with the human eye in the hope that any occurrence of disparity is noticeable. An important rule of steganography is to ensure quality degradation of the file is kept to a minimum, thus a solid steganographic application will create stego objects that look quite similar to their cover object [119]. However, when sections of the image that have not been modified during the embedding process are removed, and alternative focus is put on possible areas of message insertion in seclusion, one is quite likely to detect traces of manipulation [13].

5.1.2 Structural attacks

Quite often, the format of a digital image gets altered when an occurrence of data embedding takes places. These adjustments can indicate to a steganalyst that a form of data embedding has occurred [97]. For example, a file format such as GIF assigns 8 bits or less by constructing a palette of chosen colours. Each individual pixel of an image is defined by an index of colour within the palette. Concealing data in a GIF image by least significant bit adjustment can sometimes be unsuccessful because each palette entry is to far apart. For instance, entry 01001011 may be a dark green, whilst 01101000 may be a bright orange [119]. A lot of existing steganographic tools and techniques attempt to prevent this complication by building a different palette. An easy procedure is to select a tinier palette and duplicate the colours that are used to conceal information. However, these palettes are also easily detected, due to the presence of colour clusters within the palette. This often indicates to a steganalyst that some method of bit-twiddling has taken place. Other algorithms such as Romana Machado's, EzStego program attempts to organize the palette entries so that each entry is adjacent to a similar colour on the palette [109]. The embedding function of EzStego can be seen in Fig. 14.

Following the hiding process the palette needs to be unsorted to its initial state. If a steganalyst views the palette they will see no signs that any steganography procedure has taken place. As it stands now the information isn't stored in the least significant bits of pixels. When the recipient receives the image an identical ordering process as above in Fig. 16 must be carried out so the hidden data can be extracted by applying the new ordered indexes of the palette. The least significant bits instantly encodes the data. Nevertheless, if a steganalyst is aware of the sorting algorithm then they also will be able to access hidden bits [121, 122]. In addition, even with the above disadvantage taken into account, structural attacks are agreeably of greater importance to steganalysts as opposed to visual attacks, as they can be tested against a broader range of embedding methods [83].

5.1.3 Statistical attacks

In mathematics, the subject of statistics makes it viable to detect if any phenomenon takes place at random within a data set. Commonly, a hypothesis would be created that apparently describes why the phenomenon happens, and statistical techniques can then be used to confirm

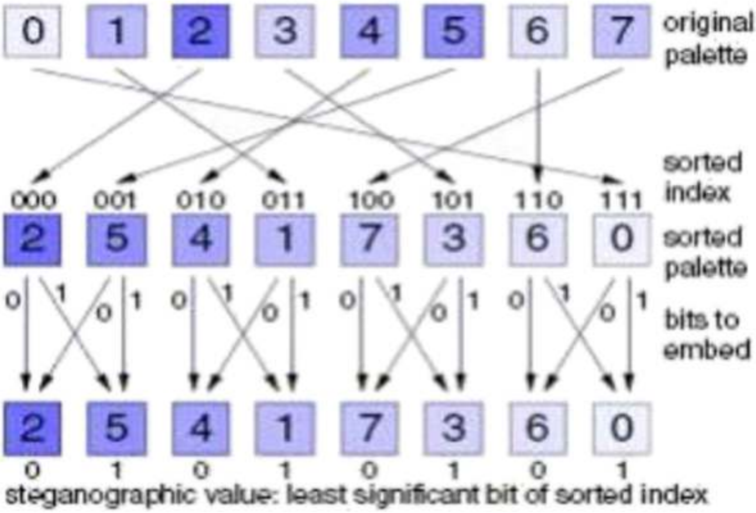


Fig. 14 EzStego embedding technique [121, 122]

this hypothesis to be either true or false. If we consider the data format for a stego object, we can start to view how statistics can be beneficial for the purpose of steganalysis, and determine whether or not an image includes secret information [23]. A stego object can be divided into two data sets, image data, and message data. The image data relates to the facts concerning the physical image that can be seen, and usually refers to pixel values. In addition, the message data refers to the facts in relation to the secret message, and if coded, it is usually more randomly constructed than image data. It can agreeably be derived that the message data is more random than image data, and this is where statistical attacks normally work. Although there is significantly less message data than image data, the tiny proportion of changeability generated by the message data is adequate enough to allow a steganalyst to invoke an attack [13]. There are many techniques recognised for determining the existence of secret data by means of statistical procedures, all directed at recognising traces of embedding for particular stego schemes. In the next section, some common statistical attacks will be discussed. The reasons as to why these attacks are so effective will also be presented.

Chi-squared (×2) test / pairs of values (POV) The Chi-squared Test, often referred to as the ×2 Test, is one of the most popular and straightforward statistical attacks in existence today. It was initially, recorded in steganalytical terms by [121, 122]. The test allows for comparison of the statistical properties (pairs of values) of a suspicious image with the theoretically anticipated statistical properties of its carrier correspondent such that it is achievable to figure out the possibility that a suspicious image is indeed a stego object [30]. For example, if we think of LSB substitution, at the time of the embedding procedure, fixed sets, of Pairs of Values (PoV): the number of 1 s and the number of 0 s show up [45]. For instance, a pixel which has an initial value of 2 would evolve into 3 if the bit to be embedded was a 1. If the bit to be embedded was a 0, the pixel would stay at 2. It was this logic, that [121, 122] used whilst developing the chi-squared attack that can be used on steganographic methods, in situations where a fixed set of PoVs are flipped into one another to embed hidden data bits. As mentioned above, this technique is established by the statistical examination of PoVs that change at the time of data embedding. When the amount of pixels for which LSB has been changed increases, both POVs frequencies tend to become the same, such that if an image contains 50 pixels which have a value 2 and 100 pixels that include a value 3. After, LSB embedding of the whole LSB plane the likely frequencies of 2 and 3 will be 75 and 75 respectively. It should be noted, that the latter is when the whole LSB plane is altered [72]. With application of the ×2 test it is not imperative for a steganalyst to have access to the cover object in order to test if data hiding has taken place, explaining why it is one of the more favourable approaches. Only in exceptional circumstance will a steganalyst have access to the original cover object, so the primary aim of the ×2 test is to be effective in establishing a technique for precisely calculating the likely statistical attributes of the initial cover object, without literally accessing it. To achieve this successfully, normally depends upon a profound understanding of numerous embedding techniques. For this reason, the test is classified as a targeted procedure. If a steganalyst is knowledgeable of a potential steganographic embedding scenario, then they are capable of analysing the significance of embedding such that they finally determine a series of features that can be examined to decide the possibility that a suspicious image is in fact a stego image [13]. Although, the above technique is popular in the detection of sequential style embedding it does not work accurately on random type embedding. Several steganographic algorithms have been created such that they randomise the embedding approach (particular algorithms include OutGuess 0.1, OutGuess 0.2, F3, F4, F5, etc.).

The extended chi-squared attack As mentioned above, it is not possible for the ×2 test to provide accurate results based on random style embedding. For example, the Chi-squared test uses an increased sample size and always starts at the beginning of an image. Due to this, changes will only be detected in the histogram if the image is distorted continuously, from start to finish thus areas of the image that are not distorted can give negative results. Whereas, the extended Chi-squared uses a constant sample size and slides the position of the samples over the entire image range, resulting in more accurate results [91]. Over the years, various efforts have been invented to generalise the concept such that it can still function. [13].The most renowned work in this area is the work carried out by [92]. As mentioned above, the procedure they used adapted the basic ×2 test by using a fixed sample size but moving the location where the samples are taken [120]. This technique is in variation to the basic ×2 test that raises the sample size and applies the test at a fixed area. It is clear that the extended approach does make it possible to detect the occurrence of randomly scattered data, yet according to [119] differentiating between embedded data and regular image data can be difficult. Bateman [13] explains that this is mainly due to the *p*-value calculation (probability that an image is a stego image) being obsolete. The p-value plot tends to rise and fall irregularly between 5% and 95%. For this reason, [13] believes that the extended chi-squared test is not proficient in the estimation the hidden message length.

Regular singular (RS) Steganalysis Another highly regarded technique for detection of LSB embedding in colour and grey-scale images was introduced by [35]. Fridrich and colleagues discuss how statistical measures on LSBs for detecting the level of embedding, alone is inaccurate. They explain that this is mainly due to the lack of unrecognisable structure of the bit plane in a stegoed image. RS Steganalysis can manipulate this feature. Fridrich et al. [35] method works by analysing embedding capacity for lossless data insertion in LSBs. Randomising LSBs minimises this capacity. To inspect an image, the authors establish two groups of fixed shape. These groups are known as Regular (R) and Singular(S) groups of pixels and are based on particular attributes. For example, whether or not the pixel noise within the group (calculated using the mean absolute value of the differences between adjacent pixels) is increased or decreased after flipping the LSBs of a fixed set of pixels within each group [62]. Subsequently, corresponding frequencies of both groups are then used to attempt to foresee the embedding degree, in the image retrieved from the initial image with flipped LSBs, and the image retrieved by randomising the LSBs of the initial image.

5.2 Blind steganalysis

In contrast to targeted steganalysis, blind steganalysis detection techniques are considerably challenging [83]. However, these methods are modern and more powerful than targeted procedures for attacking a stego file since the method does not depend on knowing any specific embedding procedures [67]. Based on this method of detection a steganalyst has no reason to think that any form of secret communications has transpired. Based on these circumstances, a series of algorithms are generally created to enable suspected files to be examined for indications of manipulations. If the algorithms indicate any evidence that tampering has occurred, then it is quite possible that the speculated file contains steganography [83]. Memon et al. [80] introduced early blind steganalysis techniques based on Image Quality Measures (IQM) were the system could easily identify images based on the possibility that they hold communicative information such as a message or a watermark. Farid [33] also introduced a technique in accordance with extracted features based on the higher order statistics (mean, variance, skewness, and kurtosis) of the wavelet (transform) of the suspected file. Farid concluded by stating that robust high-order statistical consistencies exist within the mentioned domain for natural images, and that these consistencies are modified when data is embedded. Fridrich et al. [36] contributed a more straightforward technique for blind steganalysis that was based on self-calibration. The next few sections will discuss this procedure and explain how the process makes it possible to produce an estimate of the cover image using only a suspected image file. When a steganalyst uses an estimate of the cover file it allows them to carry out more generalized attacks than prior attacks discussed in the previous sections (targeted attacks) and accurately determine any possibility that the suspect image contain message data.

5.2.1 JPEG calibration

One of the main focus points of blind steganalysis is to create an accurate estimation of the cover image. Generally, the attacks that succeed this process will measure up the statistics in the supposed cover image with that of the suspect image. A well-known method for predicting an estimate of the cover image known as JPEG calibration was proposed by Fridrich [36]. Fridrich's technique exploits the fact that many stego-systems conceal information in the transform domain at the time of the compression process. Based on the fact that the JPEG compression algorithm functions by reconstructing the image file into 8 × 8 blocks, and it is inside the indicated blocks that the encoding of the data functions, the cover work can be estimated by initiating a fresh block structure and comparing it with that of the suspect image [119]. If the outcome of the results show a big difference, this would indicate that the suspect file is likely to contain a hidden message, whereas, slight differences usually signifies that the image file does not contain a message [83]. To gain a better understanding as to how the calibration process operates a more detailed explanation of its general methodology is discussed below.

Calibration methodology The calibration procedure first will decompress the suspected image file, 4 pixels are then removed from both sides, and the result is then recompressed using the same quantization table. At this stage, the calibrated image file is still quite similar to that of the suspect file, regarding its visual and technical aspects [106]. However, by cropping and recompressing the image leads to the block structure of the suspect image being broken, this occurs because the second compression does not identify the first. Figure 15 shows a graphical representation of the embedding procedure.

Upon examination of the calibration procedure, it was discovered that cropping each aspect (top, bottom, left, and right) of the image by 4 pixels proved to be the best methodology [36]. Some research disagrees with the above mentioned cropping method and recommends that 4 pixels should be cropped from the left hand side and an additional 4 pixels cropped from the right hand side from the left-hand of the suspect image, eliminating cropping of top and bottom pixels. Yet, this technique is not deemed as efficient, as it does not eliminate the block structure as well as the latter process, i.e. the top to bottom block structure stays intact. Furthermore, cropping an image from all sides will guarantee that the whole block structure is taken out; hence a more precise estimation can be obtained [83].

Blockiness After an estimation of the cover file has been determined, the next step is to identity any existing differences in statistical properties between the calibrated image and the suspect image, and this will help to interpreted whether or not the image is a stego-image [13]. An effective technique that can be used for achieving this is known as Blockiness. The Blockiness method manipulates the fact that JPEG-driven stego-systems conceal information in the same 8 × 8 blocks that are used for compression. The technique is defined best by Dongdong Fu in [29] when it is established that: "Blockiness defines the sum of spatial discontinuities along the boundary of all 8x8 blocks of JPEG images". The philosophy behind Blockiness is that a stego image will hold a different group of coefficient's over the boundaries of each 8x8 block to that of an unstegoed image [101, 102]. As a result, the sum of the boundaries can be calculated column-wise and row-wise for both the unstegoed image and the suspect image, thus the difference between both images can be calculated (i.e. column 8 and column 9 of DCT's or pixel values). A large difference indicates that the image contains hidden data, whereas a tiny difference is most likely due to compression, and hence indicates the image is clean. The formula used for calculating the Blockiness of an image is presented in equation (Fig. 16).

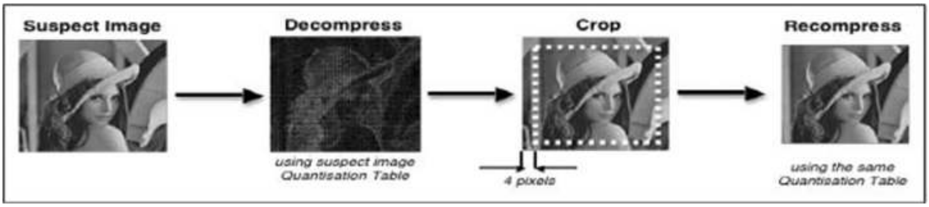

Suspect Image
Decompress
using suspect image Quantisation Table
Crop
4 pixels
Recompress
using the same Quantisation Table

Fig. 15 The calibration procedure [13]

$$B=\sum_{i=1}^{\left[\frac{M-1}{8}\right]}\sum_{j=1}^{N}\left|g_{8i,j}-g_{8i+1,j}\right|+\sum_{j=1}^{\left[\frac{N-1}{8}\right]}\sum_{i=1}^{M}\left|g_{i,8j}-g_{i,8j+1}\right|$$

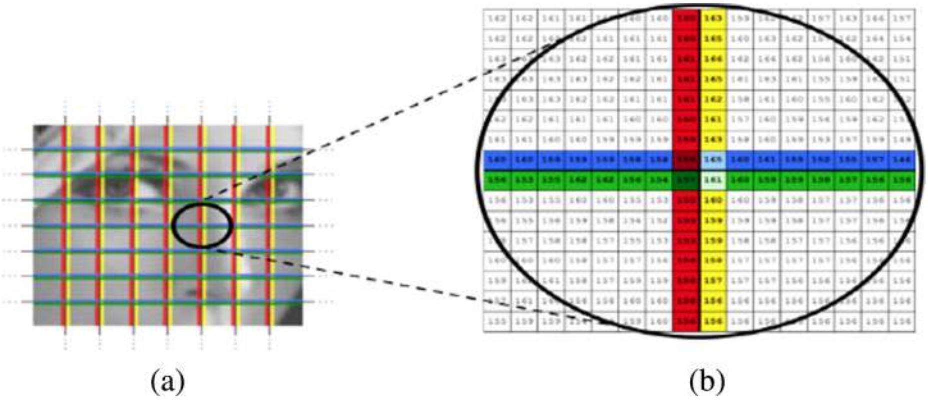
Fig. 16 Formula for calculating image blockiness

Where gi;j refers to the coordinates of a pixel value in an MxN grayscale image [125]. As seen in the equation (Fig. 16), the formula functions in a column-wise and row-wise motion instead of separately calculating the blockiness for each 8x8 block. To accomplish this, first of all, the sum of the values for the 8th row is calculated; next, the sum for its adjacent row (row 9) is calculated. The above procedure is then redone for each row-wise multiple of 8, where the each sum is added to the gathered amount until the sums of all the rows have been totalled. An identical procedure is then instantiated for the columns, before subsequently adding both totals. The result of calculating the two totals is the blockiness of the image [83]. Figure 17 shows a graphical representation of the blockiness algorithm.

Consider Fig. 17, which shows the boundaries of the 8x8 blocks in (a), and then shows how those values look in the spatial domain in (b). The red lines signify the columns that are multiples of 8, and the yellow lines display their adjacent columns that are multiples of 8 + 1. For every column, the sum of the yellow column is subtracted from the red column. Likewise, the sum of the green rows is subtracted from the blue rows. The complete values of the two separate totals are then added together to produce the blockiness value.

6 Conclusion

Over the years, many different data hiding methods have been developed. These methods can be of the spatial domain or of the transform domain with each of these domains having their own advantages and disadvantages. Nowadays, transform domain based schemes are more popular than spatial domain methods due to their higher robustness against attacks. Both, DCT and DWT are well-known transform domain methods. In recent years another transform domain technique, known as the Singular Value Decomposition has been popularly related


(a)

(b)

Fig. 17 Graphical representation of the blockiness algorithm [13]

to steganography and watermarking [60, 107]. This paper discussed targeted and blind steganalysis strategies, used for breaking steganography techniques. Both, strengths and weaknesses of these procedures were examined in relation to how simple the artefacts of message embedding can be detected by way of steganalysis. The first attack reviewed was visual attacks. It is clear that the key aspect of a productive visual attack is to accurately establish what parts of the image can be disregarded i.e. redundant data, and which parts need to be examined i.e. test data, in order to verify the theory that a suspected file has a message or watermark. However, if a steganalyst makes an incorrect judgement regarding both data types, a rise in false-negatives may occur, this is an issue that a steganalyst needs to avoid [83]. As a result, it is extremely likely that every modification of attainable redundant and test data sets is likely to be investigated so that the steganalyst is in a powerful position to make an informed judgement. For this reason, visual attacks can be tedious and time consuming. For example, the production of test images for various potential techniques of embedding would take up a lot of time. Moreover, after test images are produced they require perceptual inspection. If a steganalyst aspires to exhaust every type of embedding scenario, then thousands of images would need to be viewed to determine whether or not one suspect image is a stego image [109]. Patil et al. [83] believe that methodologies used for visual attacks are inefficient, and is generally why alternative steganalytical procedures are preferable.

Structural Attacks were also reviewed and are considered to be the more favourable approach taken by steganalyst. A Structural Attack can detect changes that may occur in an image due to data embedding, for example, changes to the palette colours/palette size, increasing or decreasing of the image size etc. If a steganalyst suspects any of the above mentioned changes, the suspected file will then be investigated further. Structural attacks can be evaluated based on a wide-range of embedding techniques. Furthermore, they more difficult from a steganographic perspective as there are likely to be a greater number of existing stego-systems where structural attacks can be practiced with success, however, more recent systems are inclined to be too secure and robust for this attack to be successful [13]. The last type of targeted attack discussed was statistical attacks. These attacks are preferred over visual or structural attacks, mainly because they can be automated. Considering this technique is capable of making an automatic analysis of the image, pressure of determining if an image is a stego image or not is taken away from the steganalyst because the analysis is done by the computer. Furthermore, automated findings will reduce the chance of misleading conclusions because of less human interpretation, unlike visual attacks. In addition, statistical attacks do not need to have an in depth knowledge of what the cover image should look like whereas, structural attacks requires the cover image to check for adjustments in image structure (i.e. palette colours) for testing [83]. However, for these attacks to work efficiently, a steganalyst must have a deep understanding of various embedding methods and have awareness as to how the stego image may have been created (referred to as a known stego-attack). If the above information is not available, then they will require access to the original image (referred to as a known-cover attack) so that differences in the original and the suspected stego can be examined.

In contrast to targeted steganalysis, blind steganalysis works based on the assumption that zero knowledge exists regarding the cover image, or the algorithm used to embed the hidden information. These attacks judge the likelihood of image tampering merely on the data contained in the suspected image. It is clear from research that blind attacks are more realistic in a real world scenario as a steganalyst is seldom knowledgeable about an image. The JPEG calibration and blockiness method shows that it is unnecessary for the cover image to be obtained for the attack to be successful. Fridrich et al. [36] noted a positive outcome with a 94% success rate. It also was successful at obtaining potential embedding strategies. Finally, as with all the steganalytical techniques explained in this thesis, the chance of success is greatly reduced when the message load is close to zero. Obviously, if only few changes are needed when the message data is hidden, fewer changes occur in the carrier file. The reason for this is that a by embedding smaller message, only a few changes will occur in the cover image, hence the stego image will look identical, or almost identical to the original image, even with hidden data embedded. Both JPEG calibration and blockiness are no different, as they too depend on message capacity, to produce a precise outcome. In addition, many trade-offs exist between the discussed techniques. For example, a stego-system that is easy to implement (i.e. LSB embedding), can also be easily attacked, whilst a more complex stego system (DCT, DWT), cannot be violated quite as easily. More complex stego-systems are inclined to be harder to break as they conceal the hidden data in a more complicated way than the simpler systems.

References

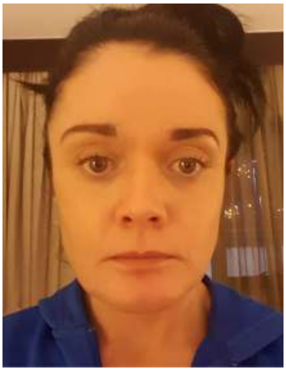
1. Abhishek N, Nandakumar K, Jain AK (2012) Multibiometric cryptosystems based on feature level fusion. IEEE Trans Inf Forensics Secur 7(1):255–1268. https://doi.org/10.1109/TIFS.2011.2166545
2. Al-Ani MS, Rajab MA (2013) Biometrics hand geometry using discrete cosine transform (DCT). Sci Technol 3(4):112–117
3. Al-Hussain A (2008) Biometric-based authentication security
4. Amritha G, Varkey M (2013) A security enhanced approach for digital image steganography using DWT and RC4 encryption. Int J Comput Trends Technol 4(6)
5. Andrews HC, Patterson CL (1976) Singular value decomposition (SVD) image coding. IEEE Trans Commun 24(4):425–432
6. Ataby A, Naima F (2010) A modified high capacity image steganography technique based on wavelet transform. Int Arab J Inform Technol 7(4)
7. Awasthi V, Tiwari KK (2012) Fingerprint analysis using termination and bifurcation minutiae. Int J Emerg Technol Adv Eng 2(2)
8. Bailey K, and Curran K (2006) An Overview of Steganography Techniques. Multimedia Tools and Applications, Vol. 30, Issue 1, ISSN: 1380-7501, Kluwer Academic, https://doi.org/10.1007/s11042-006-0008-4
9. Bandyopadhyay T, Bandyopadhyay B, Chatterjii BN (2010) Providing security of fingerprint images through digital watermarking. Proceedings of the 4th National Conference; Computing For Nation Development
10. Bansil R, Sehga S, Bedi P (2008) A novel framework for enhancing images corrupted by impulse noise using type-II fuzzy sets, in Proc. IEEE International Conference on Fuzzy Systems and Knowledge Discovery (FSKD'2008) vol. 3, pp. 266–271
11. Barnes JG (2011) Fingerprint sourcebook. CreateSpace Independent Publishing Platform, Washington, DC, pp P7–22
12. Barve S, Nagaraj U, Gulabani R (2011) Efficient and secure biometric image Stegnography using discrete wavelet transform. Int J Comput Sci Commun Netw 1
13. Bateman P (2008) Image steganography and Steganalysis. Faculty of Engineering and Physical Sciences
14. Bhattacharyya S (2012) A robust image steganography using DWT difference modulation (DWTDM). Comput Netw Inf Secur 7
15. Bhowmik P, Bhowmik K, Nurul Azam M, Wahiduzzaman Rony M (2012) Fingerprint image enhancement and it's feature extraction for recognition. Int J Sci Technol Res 1(5)
16. Cant F (2009) The secret of your child's finerprint. Dermatoglyhics.org, http://dermatoglyphics.org/11-basic-patterns-of-fingerprint/

17. Chandramouli R, Kharrazi M, Memon N (2004) Image steganography and steganalysis: concepts and practice
18. Chapman C (2010) Everything you need to know about image compression. Available: http://www.noupe.com/design/everything-you-need-to-know-about-image-compression.html. Last accessed 18th march 2014
19. Cheddad A, Condell J, Curran K, McDevitt P (2008) Biometric inspired digital image steganography. International Conference and Workshop on the Engineering of Computer Based Systems
20. Chedded A (2009) Steganoflage: a new image steganography algorithm. School of Computing & Intelligent Systems Faculty of Computing & Engineering University of Ulster
21. Chedded A, Condell J, Curran K, McDevitt P (2010) Digital image steganography: survey and analysis of current methods. Signal Process 90(3)
22. Chen PY, Ju Lin H (2006) A DWT based approach for image steganography. Int J Appl Sci Eng 4(3):275–290
23. Chhikara R, Singh L (2013) A review on digital image Steganalysis techniques categorised by features extracted. Int J Eng Innov Technol (IJEIT) 3(4)
24. Cox I, Miller M, Bloom J, Fridrich J, Kalker T (2008) Digital watermarking and steganography, 2nd edn. Kaufmann, USA, pp p1–p2
25. Cummins J, Diskin P, Lau S, Parlett R (2004) Steganography and digital watermarking
26. Currie DL, Irvine CE (1996) Surmounting the effects of lossy compression on steganography. Proceedings of the 19th National Information System Security Conference
27. Danti A, Acharya P (2010) Randomized embedding scheme based on DCT coefficients for image steganography. IJCA Special Issue on "Recent Trends in Image Processing and Pattern Recognition
28. Dass S, Jain A, Nandakumar K. (2004) Soft Biometric Traits for Personal Recognition Systems, Lecture Notes in Computer Science book series (LNCS, volume 3072), pp 731–738. https://link.springer.com/chapter/10.1007/978-3-540-25948-0_99
29. Dongdong Fu YQ, Shi DZ, Guorong X (2006) JPEG Steganalysis using empirical transition matrix in block DCT domain. Multimedia Signal Processing, IEEE 8th Workshop
30. El-Sayed M, Alfy EL, Azzat A, Sadi AL (2012) Pixel-value differencing steganography: attacks and improvements. The Second International Conference on Communications and Information Technology
31. Elysium Ltd (2007) What is the discrete wavelet transform (DWT)?.Available: http://www.jpeg.org/.demo/FAQJpeg2k/wavelet-transform.htm#What is the Discrete Wavelet Transform (DWT)?. Last accessed 19th March 2014
32. Eriksson M (2001) Biometrics fingerprint based identity verification. Thesis submitted to UMEÅ UNIVERSITY Department of Computing Science, p 29–30
33. Farid H (2002) Detecting hidden messages using higher-order statistical models. Proc. IEEE Int. Conf. Image Processing
34. Federal Bureau of Investigation (2014) Federal DNA database unit.Available: http://www.fbi.gov/. Last accessed 15th May 2014
35. Fridrich J, Goljan M, Du R (2001) Reliable detection of LSB steganography in color and gray-scale images. Mag IEEE Multimed Spec Issue Secur 8(4):22–28
36. Fridrich J, Goljan M, Hogea G (2002) Attacking the OutGuess. Proc. Multimedia and Security, Workshop at ACM Multimedia
37. Galbally J, Fierrez J, Alonso-Fernandez F, Martinez-Diaz M (2011) Evaluation of direct attacks to fingerprint verification systems. Telecommun Syst (Springer) 4-3(3–4):243–254
38. Ganic E, Eskicioglu AM (2004) Secure DWT-SVD domain image watermarking: embedding data in all frequencies. ACM Multimedia and Security Workshop
39. Ganic E, Zubair N, Eskicioglum AM (2003) An optimal watermarkeing scheme based on singular value decomposition. Proceedings of the IASTED International Conference on Communication, Network, and Information Security, p85–90
40. George JP (2012) Development of efficient biometric recognition algorithms based on fingerprint and face. A thesis submitted to the Christ University
41. Ghasemi E, Shanbehzadeh J, Fassihi N (2011) High capacity image steganography usingWavelet transform and genetic algorithm. Procedding of the International MultiConference of Engineers and Computer Science 1
42. Goel P (2008) Data hiding in digital images: a Steganographic paradigm. Department of Computer Science & Engineering Indian Institute of Technology–Kharagpur
43. Goel, V. (2017) That Fingerprint Sensor on Your Phone Is Not as Safe as You Think, New York Times, April 10th 2017. https://www.nytimes.com/2017/04/10/technology/fingerprint-security-smartphones-apple-google-samsung.html
44. Golabi S, Saadat S, Helfroush S, Tashk A (2012) A novel thinning algorithm with fingerprint minutiae extraction capability. International Journal of Computer Theory and Engineering 4(4)

45. Guillermito A (2004) A few tools to discover hidden data. Available: http://www.guillermito2.net/stegano/tools/. Last accessed 1st may 2014
46. Gunjal BL, Manthalkar RR (2010) An overview of transform domain robust digital image watermarking algorithms. Journal of Emerging Trends in Computing and Information Sciences 2(1)
47. Gupta S, Goyal A, Bhushan B (2012) Information hiding using least significant bit steganography and cryptography. Modern Education and Computer Science 6
48. Harmanpreet K, Shifali S (2014) Authenticity for protecting biometric data: facial template and iris detection. International Journal for Scientific Research & Development 2(5)
49. Hashemi AS, Zadeh HS, Ghaemmagham S, Kamarei M (2011) Universal image Steganalysis against spatial-domain steganography based on energy distribution of singular values. The 7th International Conference on Information Technology and Applications
50. Hong L, Jain A, Pankanti S, Bolle R (1997) Identity authentication using fingerprints. Proceedings of the First International Conference on Audio- and Video-Based Biometric Person Authentication 0(0):p103–p110
51. Jain AK, Uludag U (2003) Hiding biometric data, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 11, pp. 1494–1498. https://doi.org/10.1109/TPAMI.2003.1240122
52. Jain K, Nandakumar A&K (2012) Biometric Authentication: System Security and User Privacy. Computer. 45. 87–92. https://doi.org/10.1109/MC.2012.364
53. Jain A, Ross A, Prabhakar S (2004) An introduction to biometric recognition. IEEE Transactions on Circuits and Systems for Video Technology 14(1)
54. Jain AK, Bolle RM, Pankanti S (2006) Biometrics: Personal Identification in Networked Society, ISBN: 978-0-387-32659-7, https://doi.org/10.1007/978-0-387-32659-7, Springer, USA
55. Johnson NF, Jajodia S (1998a) Exploring steganography: seeing the unseen. Computer 31(2)
56. Johnson NF, Jajodia S (1998b) Steganalysis of images created using current steganography software
57. Johnson N (1999) An introduction to watermark recovery from images. Conference and Workshop on Intrusion Detection and Response (IDR'99), San Diego, California, U.S.A., 9–13 Feb. 1999, System Administration Networking Security Institute, pp. 10–24
58. Jung KH (2016) A survey of reversible data hiding methods in dual images. IETE Tech Rev 33(4):441–452
59. Jung KH (2017) A high-capacity reversible data hiding scheme based on sorting and prediction in digital images. Multimedia Tools Appl 11(76):13127–13137
60. Kamble S, Maheshkar V, Agarwal S, Srivastava VK (2012) DWT-SVD based secured image watermarking for copyright protection using visual cryptography. Computer Science & Information Technology (CS & IT)
61. Kavitha M, Kadam K, Koshti A, Dunghav P (2012) Steganography using least Signicant bit algorithm. Maharashtra Academy of Engineering, Pune university Department of Computer Engineering 2(3)
62. Ker AD (2004) Quantitative evaluation of pairs and RS steganalysis. International Society for Optics and Photonics
63. Khare K, Khare P (2010) JPEG Compression steganography & crypography using image-adaptation technique. J Advances Info Tech. 1(3):141-145, Academy Publisher, https://doi.org/10.4304/jait.1.3. 141–145
64. King R (2013) Next round of smartphones to incorporate biometrics.Available: http://www.biometricupdate.com/201303/next-round-of-smartphones-to-incorporate-biometrics. Last accessed 17th Jun 2015
65. Kocharyan D, Sarukhanyan H (2001) Feature extraction techniques and minutiae-based fingerprint recognition process. The International Journal of Multimedia Technology 1(1)
66. Koeling JM (2004) Digital imaging: a practical approach. Rowman & Littlefield, London
67. Kumar M (2011) Steganography and Steganalysis of JPEG images. LAP Lambert, USA, pp 2–4
68. Kumari M, Khare A, Khare P (2010) JPEG compression steganography & CrypographyUsing image-adaptation technique. Journal of Advances in Information Technology 1(3)
69. Lam L, Lee SW, Suen C (1992) Thinning methodologies - a comprehensive study. IEEE Trans Pattern Anal Mach Intell 14(9)
70. Lavanya N, Manjula V, Krishna Rao NV (2012) Robust and secure data hiding in image using biometric technique. International Journal of Computer Science and Information Technologies 3(5)
71. Luo W, Huang F, Huang J (2011) A more secure steganography based on adaptive pixel-value differencing scheme. Multimed Tools Appl 52(2–3):407–430
72. Lussan F (2011) A novel approach to digital watermarking, Exploiting Colour Spaces
73. Maio D, Maltoni D (1997) Direct gray-scale minutiae detection in fingerprint images. IEEE Trans Pattern Anal Mach Intell 19(1)
74. Majumder S, Devi KJ, Sarkar SK (2013) Singular value decomposition and wavelet-based iris biometric watermarking. Biometrics, IET 2(1):21–27

75. Malkhasyan N (2013) Authentication based on fingerprints with steganographic data protection. International Journal “Information Theories and Applications” 20(3)
76. Maltoni D, Maio D, Jain AK, Prabhakar S (2009a) Handbook of fingerprint recognition, 2nd edn. Spinger, London, pp 58–60
77. Maltoni D, Maio D, Jain AK, Prabhakar S (2009b) Handbook of fingerprint recognition, 2nd edn. Spinger-Verlag Limited, London, pp 57–95
78. Maltoni D, Maio D, Jain AK, Prabhakar S (2009c) Databases.Available: http://bias.csr.unibo.it/fvc2004/databases.asp. Last accessed 29th Apr 2015
79. Mayhew S (2012) Explainer: dynamic signature. Available: http://www.biometricupdate.com/201206/explainer-dynamic-signature
80. Memon N, Avcibas I, Sankur B (2001) Steganalysis based on image quality metrics. Security and Watermarking of Multimedia Contents 4314
81. Morkel T, Eloff JPH, Olivier MS (2005) An overview of image steganography. In proceedings of the fifth annual information security South Africa conference (ISSA2005), Sandton, South Africa, June/July 2005 (Published electronically)
82. O'Gorman L (1998) An overview of fingerprint verification technologies. Inf Secur Tech Rep 3(1):21–32
83. Patil K, Gupta R, Singh G (2012) Digital image steganalysis schemes for breaking steganography. International Conference on Advances in Communication and Computing Technologies
84. Patra A (2006) Development of efficient methods for face recognition and multimodal biometry. Department of Computer Science and Engineering Indian Institute of Technology Madras 0(0): p4–p5
85. Payra AK (2013) Steganology for the computer forensics examiners. LAP Lambert, Germany
86. Pennebaker W, Mitchell J (1992) JPEG: still image data compression standard, Springer, New York, ISBN-13: 978–0442012724
87. Pevny T, Fridrich J (2006) Multi-class blind Steganalysis for JPEG images. IEEE Trans. Information Forensics and Security 3(4)
88. Po WH, Yung KC, Chia YC, Hao CW (2013) Reversible data hiding algorithm using dual domain embedding. Atlantis Press
89. Prabakaran G, Bhavani R, Kanimozhi K (2013) WO secret image hiding method using SVD and DWTTECHNIQUES. International Journal of Computer Engineering and Technology (IJCET) 4(2)
90. Praveen AR (2011) Digital Image Steganography. International Journal of Computer Science & Informatics 1(11)
91. Provos N (2001) Defending against statistical steganalysis, in: proceedings of the 10th USENIX security symposium, pp. 323–336
92. Provos N, Honeyman P (2002) Detecting steganographic content on the internet. Proc. 2002 Network and Distributed System Security Symp
93. Rafizul Haque SM (2008) Singular value decomposition and discrete cosine transform based image watermarking. Interaction and System Design
94. Rakhi S (2013) Data hiding in skin tone of images using steganography. International Journal of Electronics and Communication Engineering 2(4)
95. Reddy P, Kumar S (2007) Steganalysis techniques: a comparative study. University of New Orleans Theses and Dissertations Paper 562
96. Redinbo RG, Nguyen C (2008) Overview the JPEG image compression systems. Available: http://www.ece.ucdavis.edu/cerl/ReliableJPEG/Cung/jpeg.html. Last accessed 18th march 2014
97. Rocha A, Goldenstein S (2007) Steganography and steganalysis in digital multimedia: hype or hallelujah? Journal of Theoretical and Applied Computing (RITA) 14(2)
98. Rowayda SA (2012) SVD based image processing applications: state of the art, contributions and research challenges. Int J Adv Comput Sci Appl 3(7)
99. Saha B, Sharma S (2012) Steganographic techniques of data hiding using digital images. Def Sci J 62(1): 11–18
100. Sarkar T, Sanyal S (2014). Reversible and Irreversible Data Hiding Technique . CoRR. 1405.2684
101. Schaathun HG (2012a) Machine learning in image Steganalysis. Wiley & Sons Ltd., West Sussex, pp 12–13
102. Schaathun HG (2012b) Steganalysis in the JPEG domain, in machine learning in image steganalysis. John Wiley & Sons, Ltd, Chichester. https://doi.org/10.1002/9781118437957.ch8
103. Shanthini B, Swamynathan S (2012) Multimodal biometric-based secured authentication system using steganography. Journal of Computer Science 7
104. Shejul A, Kulkarni UL (2010) A DWT based approach for steganography using biometrics. International Conference on Data Storage and Data Engineering

105. Singh AK, Mayank D, Mohan A (2013) A hybrid algorithm for image watermarking against signal processing attacks. Multi-disciplinary Trends in Artificial Intelligence 8271:235–246
106. Solanki K, Sarkar A, Manjunath BS (2007) Yass: yet another steganographic scheme that resists blind steganalysis. Information Hiding: 9th International Workshop
107. Subhedar M, Mankar VH (2015) High capacity image stegeanography based on discrete wavelet transform and singular value decomposition. International Conference on Information and Communication Technology for Competitive Strategies
108. Sukhdeep K, Manshi S (2014) Reversible data hiding and its methods: a survey. International Journal of Computer Science and Mobile Computing 3(5):p821–p826
109. Šumák M, Cmorik R (2008) Steganography detection . University of Pavol Jozef Šafárik in Košice Institute of computer science
110. Sung Liao, P Sheng Chen, T Choo Chung, P. (2001). A fast algorithm for multilevel Thresholding. J Inf Sci Eng 17, p713–727
111. Sverdlov A, Dexter S, Eskicioglu AM (2005) Robust DCT-SVD domain image watermarking for copyright protection: embedding data in all frequencies. Multimedia Computing and Networking
112. Swanirbhar M, Shaw A, Sarkar S, Sarkar SK (2013) A novel EMD based watermarking of fingerprint biometric using GEP. International Conference on Computational Intelligence: Modeling Techniques and Applications (CIMTA) 2013, Athens, Greece, pp:172–183
113. Traynor K (2014) Voice biometrics for security: the pros and cons.Available: http://www.voicetrust.com/blog/voice-biometrics-pros-and-cons/. Last accessed 17th Jun 2015
114. Uludag, U Jain, AK (2004). A case study in fingerprints. Attacks on biometric systems. USA: Morgan Kaufmann Publishers. (2002)
115. Vaghela DG, Gohil VP, Yadav R (2013) Digital watermarking: combining DCT and DWT techniques. Journal of Information, Knowledge and Research in Computer Engineering 2(2)
116. Wallhoff F (2004) FGnet – Facial expression and emotion database. Retrieved from http://citeseerx.ist.psu.edu/showciting?cid=4207225
117. Wang CM, Wu NI, Tsai CS, Hwang MS (2008) A high quality steganographic method with pixel-value differencing and modulus function. J Syst Softw 81(1):150–158
118. Wayner, P (2002) Disappearing Cryptography - Information Hiding: Steganography & Watermarking. USA: Morgan Kaufmann Publishers, ISBN: 9780123744791
119. Wayner P (2009) Disappearing cryptography information hiding:steganography and watermarking, 3rd edn. Elsevier Inc., USA, pp 344–355
120. Westfeld A (2003) Detecting low embedding rates. Information Hiding Lecture Notes in Computer Science 2578
121. Westfeld A, Pfitzmann A (1999a) Attacks on Steganographic systems breaking the Steganographic utilities EzStego, Jsteg, Steganos, and S-tools—and some lessons learned. Proceedings of the International Workshop on Information Hiding
122. Westfeld A, Pfitzmann A (1999b) Attacks on steganographic systems. Lect Notes Comput Sci 1768:61–76
123. Woodward JD, Horn C, Gatune J, Thomas A (2003) Biometrics : a look at facial recognition. RAND Public Safety and Justice
124. Wu DC, Tsai WH (2003) A steganographic method for images by pixel-value differencing. Pattern Recogn Lett 24(9):p1613–p1626
125. Xiaomei Q, Hongbin Z, Hongchen D (2007) Steganalysis for JPEG images based on statistical features of Stego and cover images. Advanced Intelligent Computing Theories and Applications. With Aspects of Theoretical and Methodological Issues Lecture Notes in Computer Science 4681(0):970–977
126. Zaheera ZA, Shibghatullah AS, Manaf M, & Anawar S (2013) Iris recognition failure in biometrics: a review. Conference: Kolokium Siswazah Sains Komputer Dan Matematik 2013 Peringkat Kebangsaan (SISKOM2013), pp: 44–58, Sweden
127. Zin WW (2013) Message embedding in PNG file using LSB Steganographic technique. International Journal of Science and Research 2(1)



Mandy Douglas recently graduated with an MSC in Computer Science from the Letterkenny Institute of Technology, Donegal, Ireland. Mandy is currently employed in the Irish IT sector. Mandy has a keen interest in steganography, image processing and other aspects of security.

Karen Bailey is a lecturer at the Institute of Technology in Letterkenny, Ireland. She holds a BSc. in Applied Sciences, a master of Philosophy and a MSc. in Computing and Information Systems. Karen is an active researcher and her research interests include Brewing, Fermentation and Starch Analysis, Steganography, Digital Watermarking, Encryption and Image Processing for Data Security and Data Hiding.

Mark Leeney is a lecturer at the Institute of Technology in Letterkenny, Ireland. He holds a PhD. in Applied maths. His research interests include Steganography, Digital Watermarking, Encryption and Image Processing for Data Security and Data Hiding.

Kevin Curran is a Reader in Computer Science and group leader for the Ambient Intelligence & Virtual Worlds Research Group at Ulster University. He is also a senior member of the IEEE. Dr. Curran has made significant contributions to advancing the knowledge of computing evidenced by over 600 published papers. He is perhaps most well-known for his work on location positioning within indoor environments, pervasive computing and Internet security. His expertise has been acknowledged by invitations to present his work at international conferences, overseas universities and research laboratories. He is a regular contributor on TV & radio and quoted in trade and consumer IT magazines on a regular basis.